Patent Application
Docket No. UF-318X
Serial No. 10/613,963

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Examiner	:	Sai Ming Chan
Art Unit	:	2616
Applicant(s)	:	Sahni *et al.*
Serial No.	:	10/613,963
Conf. No.	:	6767
Filed	:	July 3, 2003
For	:	Dynamic IP Router Using Highest Priority Matching

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

DECLARATION OF SARTAJ KUMAR SAHNI, PH.D., UNDER 37 CFR 1.132

Sir:

I, Sartaj Kumar Sahni, declare as follows:

1. I am a co-inventor of the above-identified U.S. patent application.

2. I have extensive experience in this field as evidenced by my attached *curriculum vitae*. *See Exhibit A.*

3. I am familiar with the technology described and claimed in the above-identified U.S. patent application;

4. I have reviewed and am familiar with Application Serial No. 10,613,963; the claims currently pending therein; the Office Action dated May 13, 2008; the Yazdani *et al.* ("Yazdani" U.S. Patent No. 6,859,455) and Rajasekaran *et al.* ("Rajasekaran" U.S. Pat. Pub. No. 2002/0124003) references cited therein, and the Turner *et al.* (U.S. Patent No. 6,018,524) reference cited in the Office Action dated March 19, 2007.

C:\TEMP\Declaration.doc/sjk

5. The Yazdani patent covers 3 data structures—a balanced binary search tree, a static m-way search tree, and a dynamic m-way search tree—for packet routing. The balanced binary search tree is obtained by sorting the prefixes and setting up a binary search tree for these prefixes to mirror a binary search. This is further explained in the textbook "Fundamentals of Data Structures" (Computer Science Press, Maryland, 1976) written by me, Sartaj Sahni and Ellis Horowitz, and in the textbook "Fundamentals of Data Structures in C++" (W.H. Freeman, NY, 1995) written by me, Sartaj Sahni, and Ellis Horowitz and Dinesh Mehta.

6. Referring to the Yazdani patent, the binary search tree of Figure 4 is given as an example of a search that fails to find the longest matching prefix (see col. 15, lines 3-14). The binary search tree shown in Figure 4 is not a balanced binary search tree (PTST). Rather, each node of the binary search tree shown in Figure 4 stores a prefix and not a point value as specified by the claims.

7. The term "level" is used in Yazdani *et al.* to refer to the distance from the tree root, i.e., the tree root is at level 0, its children are at level 1, their children at level 2, and so forth. In contrast, the claimed top level balanced binary search tree (PTST) has at least one node comprising a lower level range search tree (RST). This context refers to an embedding level. The entire tree constructed by each of Yazdani's methods is a top-level tree. None of the trees disclosed by Yazdani include lower-level embedded trees or other embedded structures.

8. Although the balanced binary search tree used by Yazdani can be searched for the longest matching prefix in O(log n) time, where n is the number of prefixes, it is impossible to insert or delete into/from this structure in less than linear (O(n)) time. This is because a single insert/delete can change the entire sequence of medians used to construct the tree in the first place (column 16, lines 37-56, Yazdani). Two insert procedures are described in column 18, lines 43-67 and column 19, lines 1-29 of Yazdani. Neither attempts to maintain a balanced search tree. As a result, following a series of inserts, the tree height becomes O(n) (see Sahni's data structure texts for worst-case performance of inserts/deletes that do not maintain balance) and all operations search/insert/delete run in O(n) time. The static m-way structure patented by Yazdani is a generalization of their binary tree structure and suffers from the same deficiencies as far as insert/delete are concerned. The

dynamic m-way structure uses a standard B-tree (B-trees were invented in 1972, see Sahni's data structure text for a description). However, for Yazdani's stated application, it is necessary to sort elements in subtrees and relocate them (column 21, lines 4-5). This makes the insert process inefficient. Further, no process is described to delete a prefix. As stated earlier, it is impossible to insert/delete from the Yazdani structures in O(log n) time.

9. The structure of Turner has an expected (not worst case) search complexity of O(log W), where W is the length of the longest prefix (more accurately, W may be replaced by the number of different prefix lengths). The worst-case search complexity is O(logW * log n) (provided sets of hash synonyms are maintained as balanced search trees) and so is inferior, in the worst-case, to that of Yazdani. The insert/delete complexity is $O(n \log^2 W)$ and because of the need to keep markers in each of the hash tables, it is impossible to insert/delete in O(log n) or O(log W) time using this structure.

10. As indicated above, it is impossible to get efficient insert/delete complexities using the structures of Yazdani or Turner or a combination thereof.

11. Even if it were possible to develop efficient balancing methods for the structures of Yazdani and Turner, the resulting balanced structures would be fundamentally different from those in the present patent application. For example, the Yazdani structures are single-level structures while those of the present patent are two-level structures and balancing methods do not change the number of levels (in the sense of number of levels of embedding).

12. The binary-tree on binary-tree (BOB) data structure of certain claimed embodiments in the current patent application employs red-black search trees (a popular variety of balanced binary search tree) in a very unique manner. Unlike Yazdani's search tree (which is not a red-black tree but one derived by mimicking a binary search on a sorted list), we use a binary search tree (e.g., a red-black tree) of points at the top level. Each node of this top-level red-black tree contains another red-black tree, which contains a subset of the ranges/prefixes in the router table. The total number of binary search trees in our data structure is equal to the number of node in the top-level tree plus 1. Note that Yazdani stores a single prefix in each node of a binary search tree; we store several

prefixes in each node of the top-level tree together with a single point.

13. The two data structures are very different—Yazdani is a traditional binary-search based binary search tree of prefixes, ours can be a red-black tree in which each node is itself a red-black tree.

14. To obtain the $O(\log^2 n)$ search and $O(\log n)$ insert/delete complexities claimed by us, we had to invent the idea of retaining a limited number of empty nodes in the structure (something that is not done in any other dynamic structure we are aware of). We were able to prove that even with the empty nodes counted, at most 2n nodes are needed in the red-black tree to enable the claimed complexities. Without these empty nodes, it isn't possible to get the claimed complexities.

15. Furthermore, our structure can do packet routing efficiently under any one of the following conditions—the highest priority matching prefix is to be used, the longest matching prefix is to be used, the rules are nonintersecting ranges with priority. The structures proposed in the Yazdani and Turner patents can be used only to find the longest matching prefix and cannot be extended for the other two conditions. As noted above, neither supports efficient insert/delete as supported by our structure.

16. The teachings of Rajasekaran have no relationship to our work or that of Yazdani. While Rajasekaran cite some prior art for balanced search tree structures for exact match, this prior art does not apply in any rational way to prefix matching (longest or highest priority) or to range matching and they do not attempt to address this problem.

17. Although Rajasekaran use arrays of pointers, null pointers and arrays of bits, these are standard data structure concepts with wide applicability (see any data structures book). They do not use ALLs (array linear lists, which are not the same as arrays). The bit arrays used by them are to keep track of null and non-null pointers. The claimed ALLs and bit vectors are used for entirely different purposes than the pointer and bit arrays of Rajasekaran.

18. I hereby further declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are

punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code, and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

By: ________________________ July 16, 2008
Date

SAATAJ SAHNI

C:\TEMP\Declaration.doc/sjk

EXHIBIT A

Dr. Sartaj Sahni

Distinguished Professor and Chair
CISE Dept., University of Florida, Gainesville, FL 32611
sahni@cise.ufl.edu, 352-392-1527 (voice), 352-392-1220 (fax)

LINKS

1. Wikipedia.
2. Mathematics Genealogy Project.

AREAS OF SPECIALIZATION

1. Sequential and parallel data structures and algorithms.
2. Scheduling.
3. Optimization.
4. VLSI CAD.
5. Computational geometry.
6. Image processing.
7. Medical applications.

EDUCATIONAL BACKGROUND

1. Ph.D., Computer Science, Cornell University, 1973.
2. M.S., Computer Science, Cornell University, 1972.
3. B.Tech. (Electrical Engineering), Indian Institute of Technology, Kanpur, 1970.

EMPLOYMENT

2001 - Present	Chair, CISE, University of Florida
1998 - Present	Distinguished Professor, CISE, University of Florida
1990 - 1998	Professor, CISE, University of Florida
1981 - 1990	Professor, Computer Science, University of Minnesota
1977 - 1981	Associate Professor, Computer Science, University of Minnesota
1973 - 1977	Assistant Professor, Computer Science, University of Minnesota

Ph.D. STUDENTS

1. Teofilo Gonzalez, 1975
2. Yookun Cho, 1978
3. David Nassimi, 1979
4. Eliezer Dekel, 1981
5. Raghunath Raghavan, 1982
6. Jim Cohoon, 1982

7. Ten-Hwang Lai, 1982
8. Rajiv Kane, 1984
9. Sangyong Han, 1984
10. Kam-Hoi Cheng, 1985
11. Jayaram Bhasker, 1985
12. Lishin Lin, 1986
13. Surendra Nahar, 1986
14. Jong Lee, 1987
15. Youngju Won, 1987
16. Sanjay Ranka, 1988
17. Jin Woon Woo, 1989
18. Wing Ning Li, 1989
19. San Yuan Wu, 1989
20. Patrick Jarvis, 1990
21. Jing-Fu Jenq, 1990
22. Kyunrak Chong, 1991
23. Doowon Paik, 1991
24. Mario Lopez, 1991
25. Andrew Lim, 1992
26. Keumog Ahn, 1992
27. Dinesh Mehta, 1992
28. Madhusudan Nigam, 1992
29. Venkat Thanvantri, 1995
30. Seonghun Cho, 1996
31. Chih-Fang Wang, 1998
32. Edward Cheng, 1998
33. Haejae Jung, 2000
34. Haibin Lu, 2003
35. Kun Suk Kim, 2003
36. Meongchul Song, 2005
37. Srijit Kamath, 2005
38. Joonseok Park, 2005
39. Kevin McCullen, 2006
40. Wencheng Lu, 2007
41. Xiaochun Xu, 2008

BOOKS, SOLE AUTHOR

1. *Concepts in Discrete Mathematics*, Camelot Publishing Co., Minnesota, First Edition, 1981; Second Edition, 1985, 473 pages.
2. *Software Development in Pascal*, Camelot Publishing Co., Minnesota, First Edition 1985, Second Edition 1989, Third Edition, 1993, 647 pages.
3. *Data Structures, Algorithms, and Applications in C++*, McGraw Hill, NY, 1998, 824 pages. Translated into Chinese and Greek. Web site.
   Second Edition, Silicon Press, 2005. Web site.

4. *Data Structures, Algorithms, and Applications in Java*, McGraw Hill, NY, 2000, 846 pages. Translated into Chinese. Web site.
   Second Edition, Silicon Press, 2005. Web site.

BOOKS, COAUTHORED

1. Ellis Horowitz and Sartaj Sahni, *Fundamentals of Data Structures*, Computer Science Press, Maryland, 1976, 564 pages. Translated into Portuguese.
2. Ellis Horowitz and Sartaj Sahni, *Fundamentals of Computer Algorithms*, Computer Science Press, Maryland, 1978, 626 pages. Translated into German and Japanese.
3. Ellis Horowitz and Sartaj Sahni, *Fundamentals of Data Structures in Pascal*, Computer Science Press, Maryland, First Edition 1983, Fourth Edition, 1994, 609 pages.
4. Ellis Horowitz and Sartaj Sahni, *Fundamentals of Data Structures in Turbo Pascal for the IBM-PC*, Computer Science Press, Maryland, 1988, 478 pages.
5. Sanjay Ranka and Sartaj Sahni, *Hypercube algorithms with applications to image processing and pattern recognition*, Springer-Verlag, New York, 1990, 237 pages. PDF File.
6. Ellis Horowitz, Sartaj Sahni, and Susan Anderson-Freed, *Fundamentals of Data Structures in C*, W. H. Freeman, NY, 1993, 585 pages. Translated into Korean, Italian, French, Chinese, and German. Second Edition, Silicon Press, 2007.
7. Ellis Horowitz, Sartaj Sahni, and Dinesh Mehta, *Fundamentals of Data Structures in C++*, W.H. Freeman, NY, 1995, 653 pages. Translated into Korean, French, Chinese, and German. Second edition, 2007, Silicon Press. Translated into Chinese.
8. Sartaj Sahni and Robert Cmelik, *Software Development in C*, Silicon Press, New Jersey, 1995, 553 pages.
9. Ellis Horowitz, Sartaj Sahni, and Sanguthevar Rajasekeran, *Computer Algorithms/C++*, W. H. Freeman, NY, 1997, 769 pages. Translated into Korean. French, German, and Chinese. Second Edition, Silicon Press, 2008.
10. Ellis Horowitz, Sartaj Sahni, and Sanguthevar Rajasekeran, *Computer Algorithms*, W. H. Freeman, NY, 1998, 769 pages. Second Edition, Silicon Press, 2008.
11. Sartaj Sahni and Raj Kumar, *Software Development in Java*, Silicon Press, New Jersey, 2003, 462 pages.

BOOKS/PROCEEDINGS, EDITED

1. Sartaj Sahni, *Proceedings 1987 International Conference on Parallel Processing*, Pennsylvania State University Press, PA, 1987, 993 pages.
2. Sartaj Sahni, Viktor Prasanna, and Vijay Bhatkar, *High Performance Computing*, Proceedings of the International Conference on High Performance Computing, New Delhi, India, Tata McGraw Hill, 1995, 788 pages.
3. Jose Rolim et al., *Parallel and Distributed Processing*, 15 IPDPS 2000 Workshops Proceedings, Lecture Notes in Computer Science, Volume 1800, Springer-Verlag, Berlin, 2000.

4. Sartaj Sahni, Viktor Prasanna, and Uday Shukla, *High Performance Computing--HiPC 2002*, Lecture Notes in Computer Science, Volume 2552, Springer-Verlag, Berlin, 2002.
5. Sartaj Sahni, *Proceedings of the IASTED International Conference on Computer Science and Technology*, ACTA Press, 2003 and 2004.
6. Dinesh Mehta and Sartaj Sahni, *Handbook of Data Structures and Applications*, Chapman-Hall/CRC Press, 2005.

BOOKS, CHAPTERS CONTRIBUTED

1. D. Nassimi and S. Sahni, Data Broadcasting in SIMD Computers, *Interconnection Networks for Parallel and Distributed Processing*, Wu and Feng Editors, IEEE, 1984, 282-288 (reprint).
2. D. Nassimi and S. Sahni, Parallel permutation and sorting algorithms and a new generalized connection network, *Interconnection Networks for Parallel and Distributed Processing*, Wu and Feng Editors, IEEE, 1984, 217-240 (reprint).
3. E. Dekel, D. Nassimi and S. Sahni, Parallel Matrix and Graph Algorithms, *Supercomputers: Design & Applications*, K. Hwang, Ed., IEEE, 1984, 387-403 (reprint).
4. D. Nassimi and S. Sahni, A self routing Benes network and parallel permutation algorithms, *Interconnection Networks for Parallel and Distributed Processing*, Wu and Feng Editors, IEEE, 1984, 241-249 (reprint).
5. S. Sahni, Computer Algorithms, *Encyclopedia of Physical Sciences and Technology*, Academic Press, 3, 1987, 357-375.
6. S. Sahni, A. Bhatt, and R. Raghavan, Complexity of design automation problems, *Advanced Semiconductor Technology and Computer Systems*, ed. Guy Rabbat, Von Nostrand, 1988, 526-573. PDF File.
7. S. Sahni and Z. Karian, Computer Algorithms, in *For All Practical Purposes: Introduction To Contemporary Mathematics*, COMAP, W H Freeman & Co., NY, 1988, 351-366.
8. S. Ranka and S. Sahni, Parallel algorithms for image template matching, in *Parallel Algorithms for Machine Intelligence and Vision*, eds. V. Kumar, P. Gopalakrishnan, and L. Kanal, Springer Verlag, 1990, 360-399. PDF File.
9. E. Shragowitz, J. Lee, and S. Sahni, Placer-router for sea-of-gates design style, *Progress in computer aided VLSI design*, Ed. G. Zobrist, Ablex Publishing, 2, 1990, 43-92.
10. S. Ranka and S. Sahni, Parallel algorithms for image transformations, in *Parallel algorithms and architectures for image understanding*, Ed. V. Prasanna Kumar, Academic Press, 1991, 227-248. PDF File.
11. J. Jenq and S. Sahni, Reconfigurable mesh algorithms for fundamental data manipulation operations, in *Parallel computing on distributed memory multiprocessors*, Ed. F. Ozguner, Springer Verlag, NATO ASI Series F, 1993, 27-46.
12. D. Paik and S. Sahni, Performance driven graph vertex modification problems, in *Complexity in Numerical Optimization*, World Scientific, Ed. Panos Pardolos, 1993, 299-322.

13. J. Jenq and S. Sahni, Image processing on reconfigurable meshes with buses, in *Parallel Processing for Artificial Intelligence*, North Holland, 1994, 67-91. Ed. Kanal, Kumar, Kitano, & Suttner.
14. S. Rajasekeran and S. Sahni, Fundamental algorithms for the array with reconfigurable optical buses, in *Parallel Computing Using Optical Interconnections*, Kluwer, 1998, 185-204. Ed. Li and Zhenq.
15. Chih-fang Wang and S. Sahni, OTIS optoelectronic computers, in *Parallel Computing Using Optical Interconnections*, Kluwer, 1998, 99-116. Ed. Li and Zhenq. PDF File.
16. S. Sahni, Optical and Optoelectronic Interconnection Networks, in *Advances in Switching Networks*, Kluwer, 2001, Ed. DingZhu Du and Hung Ngo. PDF File.
17. B. Vemuri, S. Sahni, F. Chen, C. Kapoor, C. Leonard, and J. Fitzsimmons, Lossless image compression. *Encyclopedia of Optical Engineering*, Ed. Ronald Driggers and Ellen Lichtenstein, Marcel Dekker Inc., 2002. PDF File.
18. S. Sahni and G. Vairaktarakis, The master-slave scheduling model, Chapter 17, 26 pages, *Scheduling: Algorithms, Models, and Performance Analysis*, Chapman-Hall/CRC Press, 2004. Ed. J. Leung.
19. S. Sahni, Analysis of algorithms, *Data Structures and Applications*, Chapman-Hall/CRC Press, 2005. Ed. D. Mehta and S. Sahni.
20. S. Sahni, Double-ended priority queues, *Data Structures and Applications*, Chapman-Hall/CRC Press, 2005. Ed. D. Mehta and S. Sahni.
21. S. Sahni, Tries, *Data Structures and Applications*, Chapman-Hall/CRC Press, 2005. Ed. D. Mehta and S. Sahni.
22. S. Sahni, Leftist Trees, *Data Structures and Applications*, Chapman-Hall/CRC Press, 2005. Ed. D. Mehta and S. Sahni.
23. S. Sahni, K. S. Kim, and H. Lu, IP Router Tables, *Data Structures and Applications*, Chapman-Hall/CRC Press, 2005. Ed. D. Mehta and S. Sahni.
24. S. Sahni, Rounding, interval partitioning and separation. *Approximation Algorithms and Metaheuristics*, Chapman-Hall/CRC Press, 2006. Ed. T. Gonzalez.
25. S. Kamath, S. Sahni, J. Palta, S. Ranka, and J. Li, Algorithms for sequencing multileaf collimaters. *Handbook of Optimization in Medicine*, Kluwer, 2006, Ed. H. Romeijn.
26. C. Wang and S. Sahni, Optical transpose systems: models and algorithms. *Handbook of Parallel Algorithms*, Chapman-Hall/CRC Press, 2007. Ed. S. Rajasekeran and J. Reif.

REFEREED JOURNAL PUBLICATIONS

1. E. Horowitz and S. Sahni, Computing Partitions with Applications to the Knapsack Problem, *Jr. of the ACM*, 21, 2, 1974, 277-292.
2. S. Sahni, Computationally Related Problems, *SIAM Jr. on Computing*, 3, 4, 1974 262-279. PDF File.
3. S. Sahni, Approximate Algorithms for the Knapsack Problem, *Jr. of the ACM*, 22, 1, 1975, 115-124.
4. E. Horowitz and S. Sahni, On Computing the Determinant of Matrices with Polynomial Entries, *Jr. of the ACM*, 22, 1, 1975, 38-50.

5. E. Horowitz and S. Sahni, The Computation of Powers of Symbolic Polynomials, *SIAM Jr. on Computing*, 4, 1, 1975, 201-208.
6. O. Ibarra and S. Sahni, Polynomially Complete Fault Detection Problems, *IEEE Trans. on Computers*, 24, 3, 1975, 242-249.
7. O. Ibarra and S. Sahni, Hierarchies of Turing Machines with Restricted Tape Alphabet Size, *JCSS*, 11, 1, 1975, 56-67.
8. S. Sahni, Algorithms for Scheduling Independent Tasks, *Jr. of the ACM*, 23, 1, 1976, 116-127.
9. E. Horowitz and S. Sahni, Exact and Approximate Algorithms for Scheduling Non-Identical Processors, *Jr. of the ACM*, 1976, 23, 2, 317-327.
10. O. Ibarra, C. Kim, and S. Sahni, Finite Automata with Multiplication, *Journal of Theoretical Computer Science*, 1976, 2, 3, 271-294
11. S. Sahni and T. Gonzalez, P-Complete Approximation Problems, *Jr. of the ACM*, 23, 3, 1976, 555-565.
12. T. Gonzalez and S. Sahni, Open Shop Scheduling to Minimize Finish Time, with *Jr. of the ACM*, 23, 4, 1976, 665-679.
13. T. Gonzalez, W.R. Franta, and S. Sahni, An Efficient Algorithm for the Kolmogorov-Smirnov and Lilliefors Tests, *ACM Trans. on Math. Software*, 3, 1, 1977, 60-64.
14. T. Gonzalez, O. Ibarra, and S. Sahni, Bounds for LPT schedules on uniform processors, *SIAM Jr. on Computing*, 6, 1, 1977, 155-166.
15. S. Sahni, General Techniques for Combinatorial Approximation, *Operations Research*, 25, 6, 1977, 920-936.
16. T. Gonzalez and S. Sahni, Preemptive Scheduling of Uniform Processors, *Jr. of the ACM*, 25, 1, 1978, 92-101.
17. T. Gonzalez and S. Sahni, Flowshop and Jobshop Schedules: Complexity and Approximation, *Operations Research*, 26, 1, 1978, 36-52.
18. T. Gonzalez, W.R. Franta, and S. Sahni, An Efficient Approximate Algorithm for the Kolmogorov-Smirnov and Lilliefors Tests, *JSCS*, 6, 1978, 257-263.
19. S. Sahni and E. Horowitz, Combinatorial Problems: Reducibility and Approximation, *Operations Research*, 26, 1978, 718-759.
20. D. Nassimi and S. Sahni, Bitonic sort on a mesh connected computer, *IEEE Trans. on Computers*, 28, 1979, 2-7.
21. S. Sahni and Y. Cho, On line scheduling of a Uniform Processor System with Release Times, *SIAM Jr. on Computing*, 8, 2, 1979, 275-285.
22. Y. Cho and S. Sahni, Complexity of scheduling shops with wait in process, *Math. Oper. Res.*, 4, 1979, 448-457.
23. S. Sahni, Preemptive Scheduling with Due Dates, *Operations Research*, 27, 5, 1979, 927-934. PDF File.
24. Y. Cho and S. Sahni, Scheduling Independent tasks with due times on a uniform Processor System, *Jr. of the ACM*, 20, 3, 1980, 550-563.
25. H. Hunt, R. Constable, and S. Sahni, On the Computational Complexity of Program Schema Equivalence, *SIAM Jr. on Computing*, 9, 2, 1980, 396-416. PDF File.
26. Y. Cho and S. Sahni, Bounds on List Schedules for Uniform Processors, *SIAM Jr. on Computing*, 9, 1, 1980, 91-103.

27. D. Nassimi and S. Sahni, An Optimal Routing Algorithm for Mesh-Connected Parallel Computers, *Jr. of the ACM*, 27, 1, 1980, 6-29.
28. D. Nassimi and S. Sahni, Finding Connected Components and Connected Ones on a Mesh-connected computer, *SIAM Jr. on Computing*, 9, 4, 1980, 744-759.
29. Y. Cho and S. Sahni, Preemptive Shop Scheduling of Independent Jobs with Release Times, *Operations Research*, 29, 3, 1981, 511-522.
30. D. Nassimi and S. Sahni, Data Broadcasting in SIMD Computers, *IEEE Trans. on Computers*, 30, 2, 1981, 101-107.
31. E. Dekel, D. Nassimi, and S. Sahni, Parallel Matrix and Graph Algorithms, *SIAM Jr. on Computing*, 10, 4, 1981, 657-675.
32. D. Nassimi and S. Sahni, A Self Routing Benes Network and Parallel Permutation Algorithms, *IEEE Trans. on Computers*, 30, 5, 1981, 332-340.
33. D. Nassimi and S. Sahni, Parallel Permutation and Sorting Algorithms and a New Generalized Connection Network, *Jr. of the ACM*, 29, 3, 1982, 642-667.
34. D. Nassimi and S. Sahni, Parallel algorithms to set up the Benes permutation network, *IEEE Trans. on Computers*, 31, 2, 1982, 148-154.
35. D. Nassimi and S. Sahni, Optimal BPC Permutations on a Cube Connected Computer, *IEEE Trans. on Computers*, 31, 4, 1982, 338-341.
36. E. Dekel and S. Sahni, Parallel scheduling algorithms, *Operation Research*, 31, 1, 1983, 24-49.
37. E. Dekel and S. Sahni, Binary trees and parallel scheduling algorithms, *IEEE Trans. on Computers*, 32, 3, 1983, 307-315.
38. R. Raghavan and S. Sahni, Single row routing, *IEEE Trans. on Computers*, 32, 3, 1983, 209-220. PDF File.
39. E. Dekel and S. Sahni, Parallel generation of prefix, postfix and tree forms, *ACM Trans. on Prog. Lang. & Systems*, 5, 3, 1983, 300-317.
40. S. Sahni and A. Bhatt, Complexity of design automation problems, *Univac Jr. of Research*, In Japanese, 1983.
41. S. Lai and S. Sahni, On line scheduling of processor systems with memories, *Jr. of Algorithms*, 4, 353-362, 1983. PDF File.
42. R. Raghavan and S. Sahni, Complexity of single row routing problems, *IEEE Trans. on Circuits and Systems*, 31, 5, 1984, 462-471.
43. S. Lai and S. Sahni, Preemptive scheduling of a multiprocessor system with memories to minimize maximum lateness, *SIAM Jr. on Computing*, 13, 4, 1984, 690-704. PDF File.
44. S. Lai and S. Sahni, Anomalies in parallel branch-and-bound algorithms, *Comm. of the ACM*, 27, 6, 1984, 594-602. PDF File.
45. E. Dekel and S. Sahni, A parallel matching algorithm for convex bipartite graphs and applications to scheduling, *Jr. of Parallel and Distributed Computing*, 1, 2, 1984, 185-205.
46. S. Sahni, Scheduling multipipeline and multiprocessor computers, *IEEE Trans. on Computers*, 33, 7, 1984, 637-645.
47. S. Han and S. Sahni, Single row routing in narrow streets, *IEEE Trans. on CAD*, 3, 3, 1984, 235-241.
48. R. Raghavan, J. Cohoon, and S. Sahni, Single bend wiring, *Jr. of Algorithms*, 7, 2, 1986, 232-257.

49. K. Cheng and S. Sahni, VLSI architectures for the finite impulse response filter, *IEEE Jr. on Selected Areas in Communications*, 4, 1, 1986, 92-99.
50. J. Bhasker and S. Sahni, A linear algorithm to determine the existence of a rectangular dual of a planar triangulated graph, *Networks*, 17, 1987, 307-317. PDF File.
51. K. Cheng and S. Sahni, VLSI architectures for bandmatrix multiplication, *Parallel Computing*, 4, 1987, 239-258.
52. J. Cohoon and S. Sahni, Heuristics for the board permutation problem, *Jr. of VLSI and Computer Systems*, 2, 1987, 37-61.
53. R. Kane and S. Sahni, A systolic design rule checker, *IEEE Trans. on CAD*, 6, 1, 1987, 22-32.
54. R. Kane and S. Sahni, Systolic algorithms for rectilinear polygons, *Computer Aided Design*, 19, 1, 1987, 15-24.
55. S. Han and S. Sahni, A fast algorithm for multilayer single row routing, *IEEE Trans. on CAD*, 6, 1, 1987, 95-102.
56. R. Kane and S. Sahni, A hardware algorithm for net extraction, with R Kane, *Computer Aided Design*, 19, 7, 1987, 347-354.
57. J. Bhasker and S. Sahni, Optimal linear arrangement of circuit components, *Jr. of VLSI & Computer Systems*, 2, 1987, 87-109. PDF File.
58. J. Bhasker and S. Sahni, A linear algorithm to find a rectangular dual of a planar triangulated graph, *Algorithmica*, 3, 1988, 247-278. PDF File.
59. S. Nahar and S. Sahni, A time and space efficient net extractor, *Computer Aided Design*, 20, 1, 1988, 17-26.
60. L. Lin, S. Sahni, and E. Shragowitz, Models and algorithms for structured layout, *Computer Aided Design*, 20, 5, 1988, 263-271.
61. L. Lin and S. Sahni, Maximum alignment of interchangeable terminals, *IEEE Trans. on Computers*, 37, 10, 1988, 1166-1177.
62. S. Sahni, Guest editor's introduction, Special issue on parallel architectures and algorithms, *Jr. of Parallel and Distributed Computing*, 5, 1988, 331-333.
63. S. Nahar and S. Sahni, A fast algorithm for polygon decomposition, *IEEE Trans. on CAD*, 7, 4, 1988, 473-483.
64. S. Wu and S. Sahni, Two NP-complete interchangeable terminal problems, *IEEE Trans. on CAD*, 7, 4, 1988, 467-472. PDF File.
65. Y. Won and S. Sahni, Maze routing on a hypercube multicomputer, *Jr. of Supercomputing*, 2, 1988, 55-79.
66. J. Lee, S. Sahni, and E. Shragowitz, A hypercube algorithm for the 0/1 knapsack problem, *Jr. of Parallel & Distributed Computing*, 5, 4, 1988, 438-456. PDF File.
67. S. Sahni, Y. Won, and S. Ranka, Programming a hypercube multicomputer, *IEEE Software*, 5, 5, 1988, 69-77. PDF File.
68. J. Lee, S. Sahni, and E. Shragowitz, Algorithms for physical design of sea-of-gates chips, *Computer Aided Design*, 20, 7, 1988, 382-397.
69. Y. Won and S. Sahni, A balanced bin sort for hypercube multicomputers, *Jr. of Supercomputing*, 2, 1988, 435-448.
70. Y. Won and S. Sahni, Hypercube-to-host sorting, *Jr. of Supercomputing*, 3, 1989, 41-61.

71. Y. Won and S. Sahni, Host-to-hypercube sorting, *Computer Systems: Science and Engineering*, 4, 3, 1989, 161-168.
72. J. Bhasker and S. Sahni, Via assignment in single row routing, *IEEE Trans. On Computers*, 38, 1, 1989, 142-148. PDF File.
73. K. Cheng and S. Sahni, VLSI architectures for the backsubstitution problem, *Parallel Computing*, 12, 1989, 53-69.
74. K. Cheng and S. Sahni, A new VLSI architecture for the adaptive recursive filtering problem, *Parallel Computing*, 10, 1989, 109-115.
75. S. Nahar, S. Sahni, and E. Shragowitz, Simulated annealing and combinatorial optimization, *Intl. Jr. of Computer Aided VLSI Design*, 1, 1, 1989, 1-23.
76. L. Lin, S. Sahni, and E. Shragowitz, An enhanced heuristic for multi-channel optimization in gate array layout, *Computer Aided Design*, 21, 2, 1989, 66-70.
77. L. Lin and S. Sahni, Fair edge deletion problems, *IEEE Trans. on Computers*, 38, 5, 1989, 756-761.
78. W. Li, S. Reddy, and S. Sahni, On path selection in combinational logic circuits, *IEEE Trans. on CAD*, 8, 1, 1989, 56-63. PDF File.
79. J. Lee, S. Sahni, and E. Shragowitz, A new approach to the function and technique of global routing, *Intl. Jr. of Computer Aided VLSI Design*, 1, 1, 1989, 25-49. PDF File.
80. J. Woo and S. Sahni, Hypercube computing: Connected components, *Jr. of Supercomputing*, 3, 209-234, 1989. PDF File.
81. Y. Won, S. Sahni, and Y. El-Ziq, A hardware accelerator for maze routing, *IEEE Trans. on Computers*, 39, 1, 1990, 141-145.
82. E. Shragowitz, J. Lee, and S. Sahni, Placer-router for sea-of-gates design style, *Progress in computer aided VLSI design*, 2, 1990, 43-92.
83. S. Ranka and S. Sahni, Image template matching on MIMD hypercube multicomputers, *Jr. of Parallel & Distributed Computing*, 10, 1990, 79-84.
84. S. Ranka and S. Sahni, Convolution on SIMD mesh connected multicomputers, *IEEE Trans. on Pattern Recognition and Machine Intelligence*, 12, 3, 1990, 315-318. PDF File.
85. S. Ranka and S. Sahni, Computing the Hough transform on hypercube multicomputers, *Jr. of Supercomputing*, 4, 1990, 169-190.
86. S. Ranka and S. Sahni, String editing on an SIMD hypercube multicomputer, *Jr of Parallel & Distributed Computing*, 9, 4, 1990, 411-418. PDF File.
87. S. Wu and S. Sahni, Covering rectilinear polygons by rectangles, *IEEE Trans. on CAD*, 9, 4, 1990, 377-388.
88. T. Gonzalez, E. Lawler, and S. Sahni, Optimal preemptive scheduling of two unrelated processors, *ORSA Jr. on Computing*, 2, 3, 1990, 219-224.
89. W. Li, S. Reddy, and S. Sahni, Long and short covering edges in combinational logic circuits, *IEEE Trans. on CAD*, 9, 12, 1990, 1245-1253. PDF File.
90. S. Ranka and S. Sahni, Odd-even shifts in SIMD hypercubes, *IEEE Trans. on Parallel & Distributed Computing*, 1, 1, 1990, 77-82. PDF File.
91. W. Li and S. Sahni, Pull-up transistor folding, *IEEE Trans. on CAD*, 9, 5, 1990, 512-521.
92. S. Ranka and S. Sahni, Clustering on an SIMD hypercube multicomputer, *IEEE Trans. on Parallel and Distributed Systems*, 2, 2, 1991, 129-137.

93. S. Ranka and S. Sahni, Efficient serial and parallel algorithms for median filtering, *IEEE Trans. on Acoustics, Speech, and Signal Processing*, 39, 6, 1991, 1462-1466. PDF File.
94. J. Woo and S. Sahni, Computing biconnected components on a hypercube, *Jr. of Supercomputing*, 5, 1, 1991, 73-87. PDF File.
95. S. Wu and S. Sahni, Fast algorithms to partition simple rectilinear polygons, *Intl. Jr. On Computer Aided VLSI Design*, 3, 1991, 241-270. PDF File.
96. J. Jenq and S. Sahni, Serial and parallel algorithms for the medial axis transform, *IEEE Trans. on Pattern and Machine Intelligence*, 14, 12, 1992, 1218-1224. PDF File.
97. W. Li, A. Lim, P. Agrawal, and S. Sahni, On the circuit implementation problem. *IEEE Trans. on CAD*, 12, 8, 1993, 1147-1156. PDF File.
98. K. Chong and S. Sahni, Optimal realizations of floorplans, *IEEE Trans. on CAD*, 12:6, 1993, 793-801.
99. K. Ahn and S. Sahni, Constrained via minimization. *IEEE Trans. on CAD*, 12, 2, 1993, 273-282. PDF File.
100. K. Chong and S. Sahni, Minimizing total wire length by flipping modules. *IEEE Trans. on CAD*, 12, 1, 1993, 167-175.
101. J. Jenq and S. Sahni, Image shrinking and expanding on a pyramid. *IEEE Trans. on Parallel and Distributed Systems*, 4, 11, 1993, 1291-1296. PDF File.
102. D. Paik and S. Sahni, Optimal folding of bit sliced stacks. *IEEE Trans. on CAD*, 12, 11, 1993, 1679-1685. PDF File.
103. A. Lim, S. Cheng, and S. Sahni, Optimal joining of compacted cells. *IEEE Trans. on Computers*, 42, 5, 1993, 597-607. PDF File.
104. K. Ahn and S. Sahni, NP-hard module rotation problems, *IEEE Trans. on Computers*, 42, 12, 1993, 1506-1510.
105. D. Mehta and S. Sahni, A data structure for circular string analysis and visualization, *IEEE Trans. on Computers*, 42, 8, 1993, 992-997.
106. M. Nigam and S. Sahni, Sorting *n* numbers on *n x n* reconfigurable meshes, *Jr. of Parallel and Distributed Computing*, 23, 1, 1994, 37-48. PDF File.
107. J. Jenq and S. Sahni, Histogramming on a reconfigurable mesh computer, *Jr. of Parallel Algorithms & Applications*, 1, 1994, 179-190. PDF File.
108. J. Jenq and S. Sahni, Reconfigurable mesh algorithms for the Hough transform, *Jr. of Parallel and Distributed Computing*, 20, 1994, 69-77. PDF File.
109. S. Sahni, Reconfigurable meshes and image processing, *Jr. of the Franklin Institute*, 1994, 3318.
110. D. Paik, S. Reddy, and S. Sahni, Deleting vertices in dags to bound path lengths. *IEEE Trans. on Computers*, 43, 9, 1994, 1091-1096. PDF File.
111. A. Lim and S. Sahni, Segmented winner trees. *Jr. Information Processing and Cybernetiks*, 30, 1, 1994, 1-15.
112. D. Mehta and S. Sahni, Computing display conflicts in string visualization, *IEEE Trans. on Computers*, 43, 3, 1994, 350-361.
113. D. Paik and S. Sahni, Network upgrading problems. *Networks*, 26, 1995, 45-58.
114. K. Chong and S. Sahni, Flipping modules to minimize the maximum wire length. *VLSI Design*, 3, 1, 1995, 37-41.

115. G. Vairaktarakis and S. Sahni, Dual criteria preemptive open shop problems with minimum finish time. *Naval Research Logistics*, 42, 1995, 103-121.
116. S. Cho and S. Sahni, Minimum area joining of compacted cells, *IEEE Trans. on CAD of ICAS*, 14, 7, 1995, 903-908. PDF File.
117. V. Thanvantri and S. Sahni, Folding a stack of equal width components. *IEEE Trans. on CAD*, 14, 6, 1995, 775-780. PDF File.
118. S. Sahni, Data manipulation on the distributed memory bus computer. *Parallel Processing Letters*, 5, 1, 1995, 3-14. PDF File.
119. S. Sahni, Scheduling master-slave multiprocessor systems, *IEEE Trans. on Computers*, 45, 10, 1996, 1195-1199. PDF File.
120. M. Lopez, R. Janardan, and S. Sahni, Efficient net extraction for restricted orientation designs, *IEEE Trans. on CAD*, 15, No. 9, 1996, 1151-1159
121. M. Nigam and S. Sahni, Sorting n^2 numbers on $n \times n$ meshes, *IEEE Trans. on Parallel & Distributed Systems*, 6, 12, 1996, 1221-1225. PDF File.
122. S. Sahni and G. Vairaktarakis, The master-slave paradigm in parallel computer and industrial settings, *Journal of Global Optimization*, 9, 1996, 357-377. PDF File.
123. V. Thanvantri and S. Sahni, Optimal folding of standard and custom cells, *ACM Trans. on Design Automation of Electronic Systems*, 1, 1, 1996, 123-143. PDF File.
124. S. Sahni and V. Thanvantri, Parallel computing: metrics and models, *IEEE Parallel and Distributed Technology*, 4, 1, 1996, 43-56. PDF File.
125. S. Rajasekaran and S. Sahni, Sorting and selection on distributed memory bus computers. *Parallel Algorithms and Applications*, 8, 1996, 179-193.
126. J. Jang, M. Nigam, V. Prasanna, and S. Sahni, Constant time algorithms for computational geometry on reconfigurable mesh, *IEEE Trans. on Parallel and Distributed Systems*, 8, 1, 1-12, 1997. PDF File.
127. A. Lim, V. Thanvantri, and S. Sahni, Planar topological routing. *IEEE Trans. on CAD*, 16, 6, 1997, 651-656. PDF File.
128. S. Rajasekaran and S. Sahni, Sorting, selection, and routing on the array with reconfigurable optical buses, *IEEE Trans. on Parallel and Distributed Systems*, 8, 11, 1997, 1133-1142. PDF File.
129. D. Mehta and S. Sahni, Models, techniques, and algorithms for finding and displaying patterns in strings and other discrete objects, *Journal of Systems and Software*, 39, 3, 1997, 201-221.
130. S. Rajasekaran and S. Sahni, Deterministic routing on the array with reconfigurable optical buses, *Parallel Processing Letters*, Dec. 1997, 219-224.
131. B. Vemuri, S. Huang, S. Sahni, C. Leonard, C. Mohr, R. Gilmore, and J. Fitzsimmons, An efficient motion estimator with application to medical image registration. *Medical Image Analysis*, 2, 1, 1998, 79-98. PDF File.
132. S. Sahni and C. Wang, BPC permutations on the OTIS-hypercube optoelectronic computer. *Informatica*, 22, 1998, 263-269. PDF File.
133. D. Paik, S. Reddy, and S. Sahni, Vertex splitting in dags and applications to partial scan designs and lossy circuits. *International Journal on Foundations of Computer Science*, 9, 4, 1998, 377-398. PDF File.

134. S. Rajasekaran and S. Sahni, Randomized routing, selection, and sorting on the OTIS-Mesh. *IEEE Trans. on Parallel and Distributed Systems*, 9, 9, 1998, 833-840.
135. S. Cho and S. Sahni, Weight biased leftist trees and modified skip lists, *ACM Jr. on Experimental Algorithmics*, Article 2, 1998. PDF File.
136. C. Wang and S. Sahni, Basic operations on the OTIS-Mesh optoelectronic computer. *IEEE Trans. on Parallel and Distributed Systems*, 9, 12, 1998, 1226-1236. PDF File.
137. S. Cho and S. Sahni, Mergeable double-ended priority queues. *International Journal on Foundations of Computer Science*, 10, 1, 1999, 1-18. PDF File.
138. E. Cheng and S. Sahni, A fast algorithm for performance-driven module implementation. *VLSI Design*, 10, 2, 1999, 237-247. PDF File.
139. C. Wang and S. Sahni, Image processing on the OTIS-Mesh optoelectronic computer. *IEEE Trans. on Parallel and Distributed Systems*, 11, 2, 2000, 97-109. PDF File.
140. K. Chong and S. Sahni, Correspondence based data structures for double ended priority queues. *ACM Jr. on Experimental Algorithmics*, Volume 5, 2000, Article 2, 22 pages. PDF File.
141. S. Sahni, Matrix multiplication and data routing using a partitioned optical passive stars network. *IEEE Trans. on Parallel and Distributed Systems*, 11, 7, 2000, 720-728. PDF File.
142. S. Sahni, The partitioned optical passive stars network: Simulations and fundamental operations. *IEEE Trans. on Parallel and Distributed Systems*, 11, 7, 2000, 739-748. PDF File.
143. S. Rajasekeran and S. Sahni, Special issue on randomized computing. *International Journal on Foundations of Computer Science*, 11, 2, 2000, 205-206.
144. S. Cho and S. Sahni, A new weight balanced binary search tree. *International Journal on Foundations of Computer Science*, 11, 3, 2000, 485-513. PDF File.
145. E. Cheng and S. Sahni, A fast algorithm for transistor folding, *VLSI Design*, 12, 1, 2001, 53-60. PDF File.
146. P. Pardalos, S. Rajasekeran, S.Sahni, and S. Shaw, Efficient algorithms for local alignment search, *Jr. of Combinatorial Optimization*, 5, 1, 2001, 117-124. PDF File.
147. S. Rajasekeran, Y. Hu, J. Luo, H. Nick, P. Pardalos, S. Sahni, and S. Shaw, Efficient algorithms for similarity search, *Jr. of Combinatorial Optimization*, 5, 1, 2001, 125-132. PDF File.
148. S. Sahni, Models and Algorithms for Optical and Optoelectronic Parallel Computers. *International Journal on Foundations of Computer Science*, 12, 3, 2001, 249-264. PDF File.
149. C. Wang and S. Sahni, Matrix multiplication on the OTIS-Mesh optoelectronic computer, *IEEE Transactions on Computers*, 50, 7, 2001, 635-646. PDF File.

150. Z. Li, I Nalcacioglu, S. Ranka, S. Sahni, J. Palta, W. Tome, and S. Kim, An algorithm for automatic, CT-based source localization after prostate implant, *Medical Physics*, 28(7), 2001, 1410-1415.

151. E. Cheng and S. Sahni, Gate resizing to reduce power consumption. *International Journal on Foundations of Computer Science*, 13, 3, 2002, 405-429. PDF File.

152. K. Kim and S. Sahni, IP lookup by binary search on length. *Journal of Interconnection Networks*, 3, 2002, 105-128. PDF File.

153. S. Kamath, S. Sahni, J. Li, J. Palta, and S. Ranka, Leaf sequencing algorithms for segmented multileaf collimation. *Physics in Medicine and Biology*, 48, 3, 2003, 307-324. PDF File. Journal Format.

154. H. Jung and S. Sahni, Supernode binary search trees. *International Journal on Foundations of Computer Science*, 14, 3, 2003, 465-490. PDF File.

155. S. Sahni, K. Kim, and H. Lu, Data structures for one-dimensional packet classification using most-specific-rule matching, *International Journal on Foundations of Computer Science*, 14, 3, 2003, 337-358. PDF File.

156. S. Sahni and K. Kim, Efficient construction of multibit tries for IP lookup. *IEEE/ACM Transactions on Networking*, 11, 4, 2003, 650-662. PDF File.

157. A. Jain, S. Sahni, J. Palta, and J. Dempsey, Partitioning 3D phantoms into homogeneous cuboids. *International Journal on Foundations of Computer Science*, 14, 5, 2003, 905-931. PDF File.

158. S. Sahni and K. Kim, O(log n) dynamic router-table design. *IEEE Transactions on Computers*, 53, 3, 2004, 351-363. PDF File.

159. G. Venkataraman, S. Sahni, and S. Mukhopadhyaya, A blocked all-pairs shortest-paths algorithm. *ACM Journal on Experimental Algorithmics*, Article 5, Volume 8, 2004.PDF File.

160. S. Kamath, S. Sahni, J. Palta, and S. Ranka, Optimal sequencing of dynamic multileaf collimators. *Physics in Medicine and Biology*, 49, 2004, 33-54. PDF File. Journal Format.

161. S. Kamath, S. Sahni, J. Palta, and S. Ranka, Optimal leaf sequencing with elimination of tongue-and-groove underdosage. *Physics in Medicine and Biology*, 49, 3, 2004, N7 - N19. PDF File. Journal Format.

162. S. Sahni and K. Kim, Efficient dynamic lookup for bursty access patterns. *International Journal on Foundations of Computer Science*, 15, 4, 2004, 567-592. PDF File.

163. S. Kamath, S. Sahni, S. Ranka, J. Li, and J. Palta, A comparison of step-and-shoot leaf sequencing algorithms that eliminate tongue-and-groove effect. *Physics in Medicine and Biology*, 49, 2004, 3137-3143. PDF File. Journal Format.

164. H. Lu and S. Sahni, O(log n) dynamic router-tables for prefixes and ranges. *IEEE Transactions on Computers*, 53, 10, 2004, 1217-1230. PDF File.

165. S. Sahni and X. Xu, Algorithms for wireless sensor networks, *Intl. Jr. on Distr. Sensor Networks, Invited Paper*, Preview Issue, 2004, 35-56 (also in 1, 1, 2005, 35-56). PDF File.

166. S. Kamath, S. Sahni, J. Li, J. Palta, and S. Ranka, Optimal field-splitting for large intensity-modulated fields. *Medical Physics*, 31, 12, 2004, 3314-3323. PDF File. Journal Format.

167. H. Lu and S. Sahni, Enhanced interval trees for dynamic IP router-tables. *IEEE Transactions on Computers*, 53, 12, 2004, 1615-1628. PDF File.
168. H. Lu, K. Kim, and S. Sahni, Prefix- and interval-partitioned dynamic IP router-tables. *IEEE Transactions on Computers*, 54, 5, 2005, 545-557. PDF File.
169. H. Lu and S. Sahni, A B-tree router-table design. *IEEE Transactions on Computers*, 54, 7, 2005, 813-824. PDF File.
170. J. Park and S. Sahni, Maximum lifetime broadcasting in wireless networks. *IEEE Transactions on Computers*, 54, 9, 2005, 1081-1090. PDF File.
171. Xuehong Sun, S. Sahni, and Yiqiang Zhao, Packet classification consuming small amount of memory. *IEEE/ACM Transactions on Networking*, 13, 5, 2005, 1135-1145. PDF File.
172. H. Lu and S. Sahni, Conflict detection and resolution in two-dimensional prefix router-tables. *IEEE/ACM Transactions on Networking*, 13, 6, 2005, 1353-1363. PDF File.
173. M. Song and S. Sahni, Approximation algorithms for multiconstrained quality-of-service routing. *IEEE Transactions on Computers*, 55, 5, 2006, 603-617. PDF File.
174. J. Park and S. Sahni, An online heuristic for maximum lifetime routing in wireless sensor networks. *IEEE Transactions on Computers*, 55, 8, 2006, 1048-1056. PDF File.
175. K. Kim and S. Sahni, Efficient construction of pipelined multibit-trie router-tables. *IEEE Transactions on Computers*, 56, 1, 2007, 32-43. PDF File.
176. H. Lu and S. Sahni, O(log W) multidimensional packet classification. *IEEE/ACM Transactions on Networking*, 15, 2, 2007, 462-472. PDF File.
177. S. Kamath, S. Sahni, J. Li, S. Ranka, and J. Palta, Generalized field splitting algorithms for optimal IMRT delivery efficiency. *Physics in Medicine and Biology*, 52, 2007, 1-14. PDF File. MedicalPhysicsWeb feature story, Choicest cuts for IMRT, on this paper.
178. X. Xu and S. Sahni, Approximation algorithms for sensor deployment. *IEEE Transactions on Computers*, 56, 2007, 1681-1695. PDF File.
179. S. Chen, M. Song, and S. Sahni, Two techniques for fast computation of constrained shortest paths. *IEEE/ACM Transactions on Networking*, 16, 1, 2008, 105-115. PDF File.
180. W. Lu and S. Sahni, Packet classification using space-efficient pipelined multibit tries. *IEEE Transactions on Computers*, 57, 5, 2008, 591-605. PDF File.
181. J. Park and S. Sahni, Power assignment for symmetric communication in wireless sensor networks. *International Journal on Distributed Sensor Networks*, to appear. PDF File.
182. W. Lu and S. Sahni, Efficient two-dimensional multibit tries for packet classification. *IEEE Transactions on Computers*, to appear. PDF File.
183. X. Xu, N. Rao, and S. Sahni, A computational geometry method for localization using differences of distances. *IEEE Transactions on Sensor Networks*, to appear. PDF File.
184. W. Lu and S. Sahni, Succinct representation of static packet classifiers. *IEEE/ACM Transactions on Networking*, to appear.

REFEREED CONFERENCE PROCEEDINGS

1. P. Goel, S. Sahni, and H. Sahasrabudhe, FORGO IV, A Beginners Fortran IV, *Proceedings of the annual meeting of the Computer Society of India*, 1970, 111-114.
2. S. Sahni, Some Related Problems from Network Flows, Game Theory and Integer Programming, *Proceedings of 13th Annual IEEE Conference on Switching and Automata Theory*, 1972, 130-138.
3. S. Sahni and T. Gonzalez, P-Complete Approximation Problems, *15th Annual IEEE Symposium on Switching and Automata Theory*, 1974, 28-32.
4. H. Hunt, R. Constable, and S. Sahni, On the Computational Complexity of Program Schema Equivalence, *8th Annual Princeton Symposium on Information Sciences and Systems*, 1974, 112-115. PDF File.
5. S. Sahni, Algorithm Design Techniques, *Proceedings Allerton Conference*, 1977, 175-185.
6. E. Dekel, D. Nassimi, and S. Sahni, Parallel Matrix and Graph Algorithms, *Proceedings Allerton Conference*, 1979, 27-36.
7. D. Nassimi and S. Sahni, Parallel Permutation and Sorting Algorithms, *Proceedings Allerton Conference*, 1979, 1-10.
8. D. Nassimi and S. Sahni, Data Broadcasting in SIMD Computers, *Proceedings 1980 IEEE International Conference on Parallel Processing*, 325-326.
9. D. Nassimi and S. Sahni, A Self Routing Benes Network and Parallel Permutation Algorithms, *Proceedings 7th ACM International Symposium on Computer Architecture*, 1980, 190-195.
10. D. Nassimi and S. Sahni, Parallel algorithms to set up the Benes permutation network, *Proceedings of the IEEE Workshop on Interconnection Networks*, 70-71, 1980.
11. S. Sahni and A. Bhatt, Complexity of design automation problems, *Proceedings of the 17th ACM/IEEE Design Automation Conference*, 1980, 402-411. *Invited Paper*. PDF File.
12. E. Dekel and S. Sahni, Parallel scheduling algorithms, *Proceedings 1981 IEEE International Conference on Parallel Processing*, 350-351.
13. E.Dekel and S. Sahni, Binary trees and parallel scheduling algorithms, *CONPAR 81*, Nurnberg, Germany, 480-492.
14. R. Raghavan, J. Cohoon, and S. Sahni, Manhattan wiring, *Proc. 19th Annual Allerton Conference*, 1981, 1-11. PDF File.
15. E.Dekel and S. Sahni, Parallel generation of the postfix form, *Proc. 1982 IEEE International Conference on Parallel Processing*, 171-177.
16. R. Raghavan and S. Sahni, Optimal single row router, *Proc. 19th ACM/IEEE Design Automation Conference*, 1982, 38-45.
17. R. Raghavan and S. Sahni, Some complexity results on the single row approach to wiring, *1982 IEEE Intl. Symp. on Circuits and Systems*, Rome, Italy, 768-771. *Invited Paper*.
18. E. Dekel and S. Sahni, A parallel matching algorithm for convex bipartite graphs and applications to scheduling, *Proc. 1982 IEEE International Conference on Parallel Processing*, 178-184.
19. J. Cohoon and S. Sahni, Heuristics for the circuit realization problem, *Proc. 20th ACM/IEEE Design Automation Conference*, 1983, 560-566.

20. S. Lai and S. Sahni, Anomalies in parallel branch-and-bound algorithms, *Proc. 1983 IEEE International Conference on Parallel Processing*, 183-190. PDF File.
21. J. Cohoon and S. Sahni, Heuristics for the board permutation problem, *Proceedings ACM/IEEE Intl. Conf. on Computer Aided Design Conference*, 1983, 223-226.
22. J. Cohoon and S. Sahni, Exact algorithms for the board permutation problem, *Proc. Allerton Conference*, 1983, 246-255.
23. S. Lai and S. Sahni, Preemptive scheduling of uniform processors with memory, *Proc. Allerton Conference*, 1983, 886-895. PDF File.
24. S. Sahni, Scheduling multipipeline and multiprocessor computers, *1984 IEEE International Conference on Parallel Processing*, 333-337.
25. R. Kane and S. Sahni, A systolic design rule checker, *1984 ACM/IEEE Design Automation Conference*, 243-250.
26. R. Kane and S. Sahni, Systolic algorithms for rectilinear polygons, *Proceedings IEEE Intl. Conf. on Computer Design*, 1984, 831-836.
27. R. Kane and S. Sahni, VLSI systems for design rule checks, *Foundations of Software Technology and Theoretical Computer Science*, Lecture Notes In Computer Science, Springer Verlag, 1984, 259-278. *Invited Paper*.
28. S. Han and S. Sahni, A fast algorithm for multilayer single row routing, *1985 ACM/IEEE Design Automation Conference*, 516-522.
29. S. Nahar, S. Sahni, and E. Shragowitz, Experiments with simulated annealing, *22nd ACM/IEEE Design Automation Conference*, 1985, 748-752.
30. R. Kane and S. Sahni, VLSI architectures for functions on rectilinear polygons, *1985 IEEE Intl. Symp. on Circuits and Systems conference*, 1203-1206.
31. R. Kane and S. Sahni, A hardware algorithm for net extraction, *1985 IEEE Intl. Symp. on Circuits and Systems conference*, 51-54.
32. S. Han and S. Sahni, A fast algorithm for single row routing, with S. Han *Proceedings 23rd Annual Allerton Conference on Communication, Control and Computing*, 1985, 324-329.
33. K. Cheng and S. Sahni, VLSI architectures for matrix multiplication, *Foundations of Software Technology and Theoretical Computer Science*, Springer Verlag, Lecture Notes In Computer Science, 206, 1985, 428-456.
34. J. Bhasker and S. Sahni, Via assignment in single row routing, *Foundations of Software Technology and Theoretical Computer Science*, Springer Verlag, Lecture Notes In Computer Science, 1986, 154-176. PDF File.
35. K. Cheng and S. Sahni, VLSI architectures for the backsubstitution problem, Information Processing 86, IFIPS Society, North Holland, 373-378. *Invited Paper*.
36. K. Cheng and S. Sahni, Multiprocessor algorithms for LU decomposition, *Proceedings 20th Annual Hawaii International Conference on System Sciences*, II, 1987, 177-187.
37. K. Cheng and S. Sahni, A new VLSI architecture for the adaptive recursive filtering problem, *1986 IEEE International Conference on Parallel Processing*, 387-389.

38. J. Bhasker and S. Sahni, A linear algorithm to obtain a rectangular dual of a planar triangulated graph, *23rd ACM/IEEE Design Automation Conference*, 108-114, 1986. PDF File.
39. S. Nahar, S. Sahni, and E. Shragowitz, Simulated annealing and combinatorial optimization, *23rd ACM/IEEE Design Automation Conference*, 1986, 293-299.
40. S. Nahar and S. Sahni, A time and space efficient net extractor, *23rd ACM/IEEE Design Automation Conference*, 1986, 411-417.
41. L. Lin, S. Sahni, and E. Shragowitz, An enhanced heuristic for multi-channel optimization in gate array layout, *Proceedings 1986 ACM/IEEE Intl. Conf. on Computer Aided Design Conference*, 242-245.
42. L. Lin, S. Sahni, and E. Shragowitz, Models and algorithms for structured layout, *Proceedings IEEE Intl. Conf. on Computer Design'86*, 346-351.
43. S. Sahni, The NTU computer science program, *ACM-IEEE Fall Joint Computer Conference*, 1986, 77-79.
44. J. Bhasker and S. Sahni, A linear algorithm to determine the existence of a rectangular dual of a planar triangulated graph, *Proceedings 20th Annual Hawaii International Conference on System Sciences*, I, 1987, 31-38. PDF File.
45. J. Bhasker and S. Sahni, Optimal linear arrangement of circuit components, *Proceedings 20th Annual Hawaii International Conference on System Sciences*, I, 1987, 99-111. PDF File.
46. E. Shragowitz, J. Lee, L. Lin, and S. Sahni, Integrated CAD system based on heterogeneous computers and multiple heuristic algorithms, *European Computer Conference*'87, 1987, 459-462.
47. Y. Won, S. Sahni, and Y. El-Ziq, A hardware accelerator for maze routing, *24th ACM/IEEE Design Automation Conference*, 1987, 800-806.
48. E. Shragowitz, J. Lee, and S. Sahni, Placer-router for sea-of-gates design style, *1987 IEEE International Conference on Computer Design*, 330-335.
49. Y. Won and S. Sahni, Maze routing on a hypercube multicomputer, *1987 IEEE International Conference on Parallel Processing*, 630-637.
50. J. Lee, S. Sahni, and E. Shragowitz, A hypercube algorithm for the 0/1 knapsack problem, *1987 IEEE International Conference on Parallel Processing*, 699-706.
51. J. Jenq and S. Sahni, All pairs shortest paths on a hypercube multiprocessor, *1987 IEEE International Conference on Parallel Processing*, 713-716. PDF File.
52. M. Nigam, D. Shetti, S. Sahni, and J. Slagle, Weapon allocation in battle, in *Expert systems and advanced data processing*, ed. Emrich, Sadlowe, and Arrowood, 1988. Proceedings of the 1987 Conference on expert systems technology in the ADP environment, 173-181.
53. S. Ranka and S. Sahni, Image template matching on SIMD hypercube multicomputers, *Proceedings 1988 International Conference on Parallel Processing*, III, Algorithms & Applications, 84-91.
54. S. Ranka and S. Sahni, Image template matching on MIMD hypercube multicomputers, *Proceedings 1988 International Conference on Parallel Processing*, III, Algorithms & Applications, 92-99. PDF File.
55. S. Ranka and S. Sahni, Convolution on SIMD mesh connected multicomputers, *Proceedings 1988 International Conference on Parallel Processing*. III, Algorithms & Applications, 212-217.

56. Y. Li, S. Reddy, and S. Sahni, On path selection in combinational logic circuits, *Proceedings 1988 ACM/IEEE Design Automation Conference*, 142-147.
57. J. Lee, Y. Won, S. Sahni, and E. Shragowitz, Parallel algorithms for physical design, *IEEE Proceedings, International Symposium on Circuits & Systems*, 1988, 325-328. *Invited Paper*.
58. J. Woo and S. Sahni, Hypercube computing: Connected components, *Proceedings IEEE workshop on future trends in distributed computing systems in the 90's*, 1988, 408-417. PDF File.
59. Y. Won and S. Sahni, Host-to-hypercube sorting, *Proceedings of International Conference on New Generation Computer Systems*, International Academic Publishers, 1989, 445-466.
60. S. Ranka and S. Sahni, Hypercube algorithms for image transformations, *Proceedings 1989 International Conference on Parallel Processing*, III-24 -- III-31. PDF File.
61. S. Ranka and S. Sahni, Efficient serial and parallel algorithms for median filtering, *Proceedings 1989 International Conference on Parallel Processing*, III-56 -- III-62. PDF File.
62. S. Ranka and S. Sahni, Clustering on an SIMD hypercube multicomputer, *Proceedings 10th IEEE International Conference on Pattern Recognition*, 1990, 532-536.
63. J. Woo and S. Sahni, Computing biconnected components on a hypercube, *Proceedings, Second IEEE Workshop on Future Trends of Distributed Computing Systems*, 1990, 277-283. PDF File.
64. M. Nigam, S. Sahni, and B. Krishnamurthy, Mapping Hamiltonians and hypercubes in star graphs, *Proceedings 1990 International Conference on Parallel Processing*, 340-343.
65. J. Woo and S. Sahni, Load balancing on a hypercube multicomputer, *Fifth IEEE International Parallel Processing Symposium*, 1991, 525-530.
66. J. Jenq and S. Sahni, Reconfigurable mesh algorithms for the Hough transform. *Proceedings 1991 International Conference on Parallel Processing*, 3, 34-41. PDF File.
67. J. Jenq and S. Sahni, Reconfigurable mesh algorithms for image shrinking, expanding, clustering, and template matching. *Fifth IEEE International Parallel Processing Symposium*, 1991, 208-215. PDF File.
68. K. Chong and S. Sahni, Flipping modules to minimize the maximum wire length. *Proc. IEEE Intl. Conf. on Computer Design, ICCD'91*, 528-531.
69. A. Lim, S. Cheng, and S. Sahni, Optimal joining of compacted cells. *Proceedings 1991 MIT Conference on Advanced VLSI Design*, 99-112.
70. J. Jenq and S. Sahni, Reconfigurable mesh algorithms for the area and perimeter of image components. *Proceedings 1991 International Conference on Parallel Processing*, vol 3, 280-281. PDF File.
71. P. McGeer, R. Brayton, S. Sahni, and A. Sangiovanni-Vincentelli, Performance enhancement through the generalized bypass transform. *Proceedings ACM/IEEE Intl. Conf. on Computer Aided Design 91*, 232-235.
72. D. Paik and S. Sahni, Graph vertex modification problems and applications, *Invited Paper*, *Proceedings of the Sixth International Symposium on Computer*

and Information Sciences, Turkey, 1991, 381-394, Elsevier, and *1991 Great Lakes Symposium*, Michigan, Keynote Speaker.

73. J. Jenq and S. Sahni, Serial and parallel algorithms for the medial axis transform, *Proceedings IEEE International Parallel Processing Symposium*, 1992, 326-333. PDF File.
74. J. Jenq and S. Sahni, Histogramming on a reconfigurable mesh computer, *Proceedings IEEE International Parallel Processing Symposium*, 1992, 425-432. PDF File.
75. W. Li, A. Lim, P. Agrawal, and S. Sahni, On the circuit implementation problem. *Proceedings ACM/IEEE Design Automation Conference*, 1992, 478-483. PDF File.
76. J. Jenq and S. Sahni, Image shrinking and expanding on a pyramid. *Proceedings 1992 International Conference on Parallel Processing*, III-302--III-309. PDF File.
77. K. Chong and S. Sahni, Minimizing total wire length by flipping modules. *Fifth IEEE International Conference On VLSI Design*, 1992, 25-30.
78. D. Mehta and S. Sahni, Computing display conflicts in string and circular string visualization. In *Combinatorial Pattern Matching*, Lecture Notes in Computer Science, Springer Verlag, 644, 1992, 244-261.
79. D. Mehta and S. Sahni, Techniques and models for the visualization of labeled discrete objects. *1992 ACM Symposium on Applied Computing*, 1224-1233.
80. A. Lim and S. Sahni, Segmented winner trees. *Proceedings, 30th Annual ACM Southeastern Conference*, 1992, 67-71.
81. D. Paik, S. Reddy, and S. Sahni, Heuristics for the placement of flip-flops in partial scan designs and for the placement of signal boosters in lossy circuits. *Sixth IEEE International Conference On VLSI Design*, 45-50, 1993.
82. M. Lopez, R. Janardan, and S. Sahni, A fast algorithm for net extraction. *Proceedings ACM/IEEE Intl. Conf. on Computer Aided Design 93*, 770-774.
83. M. Nigam and S. Sahni, Sorting n numbers on $n \times n$ reconfigurable meshes, *Seventh IEEE International Parallel Processing Symposium*, 1993, 174-181. PDF File.
84. M. Nigam and S. Sahni, Sorting n^2 numbers on $n \times n$ meshes, *Seventh IEEE International Parallel Processing Symposium*, 1993, 73-78. PDF File.
85. M. Nigam and S. Sahni, Parallel programming on reconfigurable meshes with buses, *Proceedings Fourth IEEE Workshop on Future Trends of Distributed Computing Systems*, 1993, 188-193.
86. Keumog Ahn and S. Sahni, Flipping modules to improve circuit performance and routability, *Proc. 7th IEEE International Conf. on VLSI Design*, 1994, 127-132.
87. M. Nigam and S. Sahni, Computational geometry on a reconfigurable mesh, *Eighth IEEE International Parallel Processing Symposium*, 1994, 86-93.
88. M. Nigam and S. Sahni, Triangulation on a reconfigurable mesh with buses, *International Conference on Parallel Processing*, 1994, 251-257. PDF File.
89. V. Thanvantri and S. Sahni, Folding a stack of equal width components. *Proceedings 1994 ACM/IEEE Intl. Conf. on Computer Aided Design*, 432-435. PDF File.

90. S. Sahni, Computing on reconfigurable bus architectures, *Computer Systems & Education*, Editors: Balakrishnan et al., Tata McGraw-Hill Publishing Co., New Delhi, 1994, 386-398. *Invited state-of-the-art paper*. PDF File.
91. A. Lim, S. Sahni and V. Thanvantri, A fast algorithm to test planar topological routing. *Proceedings IEEE VLSI Design 1995*, 8-12. PDF File.
92. S. Sahni, Reconfigurable meshes and image processing, in *Parallel and Distributed Signal and Image Integration Problems*, Ed. Madan et al., World Scientific, 1995, 64-81.
93. S. Sahni, Scheduling master-slave multiprocessor systems, *Proceedings, First International EURO-PAR Conference*, Lecture Notes In Computer Science, Vol. 966, Springer, 1995, 611-622.
94. S. Rajasekaran and S. Sahni, Sorting and selection on distributed memory bus computers. *Proc. 1995 International Conference on Parallel Processing*, III-151---III-154.
95. S. Rajasekaran and S. Sahni, Sorting and routing on the array with reconfigurable optical buses, *1996 IEEE Second International Conference on Algorithms & Architectures for Parallel Processing, ICA\u3\dPP*, 217-224. PDF File.
96. S. Cho and S. Sahni, Weight biased leftist trees and modified skip lists, *Proceedings, Second International Conference, COCOON'96*, Lecture Notes in Computer Science, Springer Verlag, 1090, 361-370.
97. B. Vemuri, S. Huang, S. Sahni, C. Leonard, C. Mohr, T. Lucas, R. Gilmore, and J. Fitzsimmons, A robust and efficient algorithm for image registration, *XVth Intl. Conference on Information Processing in Medical Imaging (IPMI)*, 1997, 465-470.
98. S. Sahni and Chih-Fang Wang, BPC permutations on the OTIS-mesh optoelectronic computer. *IEEE Conference on Massively Parallel Programming with Optical Interconnect*, 1997, 130-135. PDF File.
99. S. Rajasekeran and S. Sahni, Computing on the array with reconfigurable optical buses, *World Multiconference on Systemics, Cybernetics, and Informatics*, Caracas, Venezuela, 1997, 459-466.
100. S. Sahni and G. Vairaktarakis, Scheduling for distributed computing, *Proceedings IEEE 6th Workshop on Future Trends of Distributed Computing Systems*, 1997, 284-289. PDF File.
101. C. Wang and S. Sahni, Basic operations on the OTIS-Mesh optoelectronic computer. *IEEE Conference on Massively Parallel Programming with Optical Interconnect*, 1998, 150-157. PDF File.
102. S. Rajasekaran and S. Sahni, Randomized routing, selection, and sorting on the OTIS-Mesh. *Second IASTED International Conference, European Parallel and Distributed Systems (Euro-PDS'98)*, 1998, 111-116.
103. F. Chen, S. Sahni, and B. C. Vemuri, Efficient algorithms for lossless compression of 2D images. *Workshop on Data Compression Processing Techniques for Missile Guidance Links*, US Army Aviation and Missile Command, 1998, 811-829.
104. H. Jung, J. Palta, S. Ranka, S. Sahni, T. Zhu. Fast Hierarchical Algorithm for Photon Beam Dose Calculation, {\em 40th Annual Meeting of the American Association of Physics in Medicine}, 1998.

105. X. Fang, Z. Li, J. Palta, S. Ranka, and S. Sahni. Fast Hierarchical Algorithms for Brachytherapy Calculation Using Monte Carlo Simulation, {\em 40th Annual Meeting of the American Association of Physics in Medicine}, 1998.
106. B. C. Vemuri, S. Huang, S. Sahni, T. Leonard and J. Fitzsimmons, Reliable and efficient image registration. *Indian conf. on Compu. Vision, Graphics and Image Processing (ICVGIP)*, New Delhi, India, Dec. 21-23, 1998, pp. 109-114. PDF File.
107. F. Chen, S. Sahni, and B. C. Vemuri, Efficient algorithms for lossless compression of 2D/3D images. *Third International Conference on Visual Information Systems*, Springer Verlag, vol. 1614, 1999, pp. 681-688. PDF File.
108. S. Sahni, B. Vemuri, F. Chen, and C. Kapoor, Variable-bit-length coding: An effective coding method. *Third International Conference on Visual Information Systems*, Springer Verlag, vol. 1614, 1999, pp. 665-672.
109. S. Sahni, Models and Algorithms for Optical and Optoelectronic Parallel Computers. *International Symposium on Parallel Architectures, Algorithms and Networks*, IEEE Computer Society Press, 1999, pp. 2-7.
110. J. In, C. Jin, J. Peir, S. Ranka, and S. Sahni, A framework for matching parallel applications with parallel machines, *6th International Conference on High Performance Computing*, Lecture Notes in Computer Science, Springer Verlag, Volume 1745, 1999, 331-338. PDF File.
111. C. Wang and S. Sahni, Matrix multiplication on the OTIS-Mesh optoelectronic computer, *6th IEEE International Conference on Parallel Interconnects*, 1999, 131-138. PDF File.
112. G. Venkataraman, S. Sahni, and S. Mukhopadhyaya, A blocked all-pairs shortest-paths algorithm. *Scandinavian Workshop on Algorithms and Theory*, Lecture Notes in Computer Science, Vol. 1851, Editor: Magnus Halldorsson, Springer Verlag, 2000, 419-432. PDF File.
113. S. Sahni and K. Kim, Efficient Construction Of Fixed-Stride Multibit Tries For IP Lookup, *Proceedings 8th IEEE Workshop on Future Trends of Distributed*, 2001, 178-184. PDF File.
114. S. Sahni and K. Kim, Efficient Construction Of Variable-Stride Multibit Tries For IP Lookup, *Proceedings IEEE Symposium on Applications and the Internet, SAINT*, 2002, 220-227. PDF File.
115. S. Sahni, K. Kim, and H. Lu, Data structures for one-dimensional packet classification using most-specific-rule matching, *International Symposium on Parallel Architectures, Algorithms and Networks*, 2002, 3-14. PDF File.
116. S. Sahni and K. Kim, O(log n) dynamic packet routing. *IEEE Symposium on Computers and Communications*, 2002, 443-448. PDF File.
117. C. Wang and S. Sahni, Computational geometry on the OTIS-mesh optoelectronic computer, *International Conference on Parallel Processing*, 2002, 501-507. PDF File.
118. K. Kim and S. Sahni, IP lookup by binary search on length. *IEEE Symposium on Computers and Communications*, 2003, 77-82. PDF File.
119. H. Lu and S. Sahni, O(log n) dynamic router-tables for ranges. *IEEE Symposium on Computers and Communications*, 2003, 91-96. PDF File.

120. H. Lu and S. Sahni, Dynamic IP router-tables using highest-priority matching. *IEEE Symposium on Computers and Communications*, 858-863, 2004. PDF File.

121. H. Lu and S. Sahni, A B-tree router-table design. *IEEE Symposium on Computers and Communications*, 840-845, 2004. PDF File.

122. Xuehong Sun, S. Sahni, and Yiqiang Zhao, Fast update algorithm for IP forwarding table using independent sets. *IEEE International Conference on High Speed Networks and Multimedia Communications*, 2004. PDF File.

123. S. Kamath, S. Sahni, J. Palta, S. Ranka, and J. Li, Optimal Leaf Sequencing with Elimination of Tongue-and-Groove Underdosage, *46th Annual Meeting of the American Association of Physicists in Medicine*, 2004, 1845.

124. S. Kamath, J. Li, S. Sahni, S. Ranka, and J. Palta, Optimal Field Splitting for Large Intensity-Modulated Fields, *46th Annual Meeting of the American Association of Physicists in Medicine*, 2004, 1906-1907.

125. H. Lu, K. Kim, and S. Sahni, Prefix- and interval-partitioned dynamic IP router-tables. *IEEE Globecom*, 2004. PDF File.

126. S. Chen, M. Song, and S. Sahni, Two techniques for fast computation of constrained shortest paths. *IEEE Globecom*, 2004. PDF File.

127. S. Kamath, S. Sahni, S. Ranka, J. Li, and J. Palta, IMRT Leaf sequencing algorithms, *Book of Invited Talks, Silver Jubilee AMPI Conference, International Conference on Medical Physics* ICMP 2004, 50-53. *Invited Paper*.

128. J. Park and S. Sahni, Maximum lifetime broadcasting in wireless networks. *ACS/IEEE Intl. Conf. on Computer Systems and Applications (AICCSA)*, 2005. *Invited Paper*. PDF File.

129. W. Lu and S. Sahni, Packet classification using two-dimensional multibit tries. *IEEE Symposium on Computers and Communications*, 2005, 849-854. PDF File.

130. W. Lu and S. Sahni, Packet forwarding using pipelined multibit tries. *IEEE Symposium on Computers and Communications*, 2006. PDF File.

131. W. Lu and S. Sahni, Packet classification using pipelined two-dimensional multibit tries. *IEEE Symposium on Computers and Communications*, 2006. PDF File.

132. J. Park and S. Sahni, Power assignment for symmetric communication in wireless sensor networks. *IEEE Symposium on Computers and Communications*, 2006. PDF File.

133. X. Xu and S. Sahni, Approximation algorithms for sensor deployment. *Innovations and Real-Time Applications of Distributed Sensor Network (DSN) Symposium*, 2006. *Best Paper Award*. PDF File.

134. H. Lu and S. Sahni, Dynamic tree bitmap for IP lookup and update. *International Conference on Networking*, 2007. PDF File.

135. S. Sahni, N. Rao, S. Ranka, Y. Li, E. Jung, and N. Kamath, Bandwidth scheduling and path computation algorithms for connection-oriented networks. *International Conference on Networking*, 2007. *Best Paper Award*. PDF File.

136. W. Lu and S. Sahni, Succinct representation of static packet forwarding tables. *International Conference on Networking*, 2007. PDF File.

137. W. Lu and S. Sahni, Recursively partitioned static IP router-tables. *IEEE Symposium on Computers and Communications*, 2007. PDF File.

138. W. Lu and S. Sahni, Succinct representation of static packet classifiers. *IEEE Symposium on Computers and Communications*, 2007. PDF File.

139. N. Rao, X. Xu, and S. Sahni, A computational geometry method for DTOA triangulation. *Fusion*, 2007. PDF File.

140. W. Lu and S. Sahni, Low power TCAMs for very large forwarding tables. *INFOCOM*, 2008.

141. E. Jung, Y. Li, S. Ranka, and S. Sahni, An evaluation of in-advance bandwidth scheduling algorithms for connection-oriented networks. *International Symp. on Parallel Architectures, Algorithms, and Networks (ISPAN)*, 2008. PDF File.

142. X. Zha and S. Sahni, Highly compressed Aho-Corasick automata for efficient intrusion detection. *IEEE Symposium on Computers and Communications*, 2008.

143. E. Jung, Y. Li, S. Ranka, and S. Sahni, Performance evaluation of routing and wavelength assignment algorithms for optical networks. *IEEE Symposium on Computers and Communications*, 2008.

144. X. Xu, S. Sahni, and N. Rao, Minimum cost sensor coverage of planar regions. *Fusion*, 2008.

145. X. Xu, S. Sahni, and N. Rao, On basic properties of localization using distance-difference measurements. *Fusion*, 2008.

146. N. S. V. Rao, M. Shankar, J. Chin, D. Yau, Y. Yang, J. Hou X. Xu, and S. Sahni, Localization under random measurements with application to radiation sources. *Fusion*, 2008.

147. S. Rajasekaran, V. Kumar, S. Sahni, and R. Birge, Efficient algorithms for protein-based associative processors and volumetric memories. *8th IEEE Conference on Nanotechnology*, 2008.

PAPERS IN REVIEW

1. X. Xu, S. Sahni, and N. Rao, Sensor placement for q-coverage of planar regions.

PATENTS & COPYRIGHTS

1. H. Lu and S. Sahni, Nesting ranges. Copyright registration Txu 1-042-495. Registration date May 28, 2002.
2. H. Lu and S. Sahni, Conflict free ranges. Copyright registration Txu 1-042-496. Registration date May 28, 2002.
3. H. Lu and S. Sahni, Dynamic IP router-tables using highest-priority matching. Copyright registration Txu 1-086-569. Registration date Nov. 20, 2002.
4. J. Park and S. Sahni, Maximum lifetime broadcasting in wireless networks. Copyright registration Txu 1-282-550. Registration date Jan 30, 2006.
5. H. Lu and S. Sahni, Conflict detection and resolution in two dimensional prefix router tables. Copyright registration Txu 1-308-038. Registration date June 14, 2006.

6. S. Kamath, S. Sahni, J. Palta, S. Ranka, and J. Li, Leaf sequencing method and system. U.S Patent No. 7,085,348, August 1, 2006.
7. S. Kamath, S. Sahni, J. Li, J. Palta, and S. Ranka, Field splitting for intensity-modulated fields of large size. U.S Patent No. 7,142,635 B2, Nov. 28, 2006.
8. H. Lu and S. Sahni, O(log n) dynamic router tables for prefixes and ranges. Patent filed April 30, 2003, Serial Number 10/426,423.
9. H. Lu and S. Sahni, Dynamic IP router-tables using highest-priority matching. Patent filed July 3, 2003, Serial Number 10/613,963. International application number PCT/US03/21135.
10. H. Lu and S. Sahni, B-tree dynamic router-table design. Patent filed Dec. 31, 2003.
11. H. Lu and S. Sahni, Prefix partitioning methods for dynamic IP router-tables. Patent filed Nov. 21, 2003, Serial Number 10/719,914.
12. H. Lu, K. Kim, and S. Sahni, Partitioning methods for dynamic IP router-tables. Patent filed Nov. 21, 2003, Serial Number 10/718,842.
13. H. Lu and S. Sahni, Apparatus and method for detecting and resolving conflicts in a data communication network. Patent filed July 8, 2004, Serial Number 60/586,386.
14. H. Lu and S. Sahni, Multidimensional packet classification for internet routers and firewalls. Patent filed July 8, 2004, Serial Number 60/586,383. Copyright TX1-282-490, Jan. 30, 2006.
15. K. Kim and S. Sahni, Efficient construction of pipelined multibit-trie router-tables. Patent filed July 13, 2004.
16. W. Lu and S. Sahni, Packet classification using two-dimensional multibit tries. Patent filed Nov. 5, 2004.
17. W. Lu and S. Sahni, Packet classification using pipelined multi-bit tries, Copyright TXu 1-293-394, Feb. 28, 2006.
18. S. Kamath, S. Sahni, J. Li, J. Palta, and S. Ranka, Variable feathering field splitting for intensity modulated fields of large size. July, 2005.
19. W. Lu and S. Sahni, Succinct representation of static packet classifiers. 60/821,220. 2006.
20. H. Lu and S. Sahni, Dynamic tree bitmap for IP lookup and update. 2006.
21. S. Sahni, N. Rao, S. Ranka, Y. Li, E. Jung, and N. Kamath, Method and systems for bandwidth scheduling and path computation for connection-oriented networks.

EDITOR OF A SCHOLARLY JOURNAL

1. Area editor, Data structures & Algorithms, *Jr of Parallel & Distributed Computing*, 1984-86.
2. Area Editor, Algorithms for Multiprocessors, *Jr of Parallel & Distributed Computing*, 1986-92.
3. Member, Editorial Advisory Board, *Information & Software Technology*, Butterworth Scientific Ltd. 1986-1994.
4. Member, Editorial Advisory Board, *Computer Systems: Science and Engineering*, Butterworth Scientific Ltd. 1988-.
5. Associate Editor, *IEEE Transactions on Parallel and Distributed Systems*. 1991-1994.

6. Co-Editor, Theory, Algorithms, and Programming, *Jr of Parallel & Distributed Computing*, 1992-.
7. Co-Editor, *Parallel Computing Series*, Chapman-Hall, England, 1992-1994.
8. Associate Editor, *IEEE Parallel and Distributed Technology: Systems and Applications*, 1992-96.
9. Associate Editor, *IEEE Concurrency*, 1996-97.
10. Member, Editorial Board, *International Journal of Foundations of Computer Science*, 1997-1999.
11. Managing Editor, *International Journal of Foundations of Computer Science*, 1999-.
12. Member, Editorial Board, *Parallel Processing Letters*, 2001-.
13. Editor-in-Chief, *Computer and Information Science Series*, Chapman & Hall/CRC, 2002-.
14. Member, Editorial Board, *International Journal of Computational Science and Engineering*, 2003-.
15. Member, Editorial Board, *International Journal of High Performance Computing and Networking*, 2004-.
16. Member, Editorial Board, *International Journal of Distributed Sensor Networks*, 2004-.
17. Member, Advisory Board, *International Journal of Pervasive Computing and Communications*, 2004-.
18. Member, International Advisory Board, *Sultan Qaboos University Journal for Science*, 2005-.
19. Member, Editorial Advisory Board, *Enterprise Information Systems*, Taylor and Francis. 2006-.
20. Member, Editorial Board, *Lecture Notes on ICST Activities* (Institue for Computer Sciences, Social-Informatics, and Telecommunications Engineering), Springer Verlag, 2008-.

PROFESSIONAL ACTIVITIES

1. Member, Conference advisory committee, Foundations of software technology and theoretical computer science, India, 1980-1990.
2. Chairman, Computer Science Curriculum, National Technological University, Colorado, 1983-.
3. Member, Advisory Board, Bilkent University, Ankara, Turkey, 1987.
4. Program Chair, 1987 IEEE International Conference on Parallel Processing.
5. General Chair, 1991 IEEE Symposium on Parallel and Distributed Processing.
6. Steering Committee Member, IEEE Symposium on Parallel and Distributed Processing (SPDP), 1991 and 1995.
7. Steering Committee Member, International Conference on High Performance Computing (HiPC), 1995-.
8. Member, Steering Committee, International Parallel Processing Symposium (IPPS) (renamed IPDPS in 1999), 1996-.
9. Member, Advisory Committee, International Symposium on Parallel Architectures, Algorithms and Networks, 1997-.

10. Member, Advisory Committee, Workshop on Randomization Methods in Algorithm Design, 1997-.
11. Member, Steering Committee, Workshop on Biologically Inspired Solutions to Parallel Processing Problems (BioSP3), 1997-.
12. Keynote Speaker, Second Great Lakes Computer Science Conference, Western Michigan University, Kalamazoo, Oct. 17, 1991.
13. Keynote Speaker, 5th Workshop on Algorithmic Research in the Midsouth, Southwest Louisiana University, Lafayette, April 24, 1992.
14. Member, Advisory Committee, IEEE Technical Committee on Parallel Processing, 1993-96.
15. Chair, IEEE Technical Committee on Parallel Processing, 1996-1999.
16. Vice Chair, IEEE Technical Committee on Parallel Processing, 1999-.
17. Program Vice-chair for algorithms and applications, 8th International Parallel Processing Symposium, 1994.
18. Program Chair, Systems Track, IEEE Symposium on Parallel and Distributed Processing, 1995.
19. Keynote Speaker, International Parallel Processing Symposium, 1995.
20. Program Chair, International Conference on High Performance Computing, 1995-1997.
21. Program Vice-chair for algorithms, International Parallel Processing Symposium, 1997.
22. Program Chair, International Parallel Processing Symposium, 1998.
23. Keynote Speaker, International Symposium on Parallel Architectures, Algorithms, and Networks, 1999.
24. Keynote Speaker, IEEE International High Performance Computing Conference, 1999.
25. Poster/presentation Chair, IEEE International Conference on High Performance Computing, 1999-2001.
26. Keynote Speaker, Hawaiian International Conference on System Sciences, 2000.
27. Member, Steering committee, Workshop on Advances in Parallel and Distributed Computational Models, 2000-.
28. General Chair, Seventh International Workshop on Solving Irregularly Structured Problems in Parallel, 2000.
29. Program Co-chair, 7th IEEE International Conference on Parallel Interconnect, 2000.
30. Member, Advisory Committee, Emerging Technology Track, Hawiian International Conference on System Sciences (HICSS34), 2001.
31. Keynote speaker, Workshop on Advances in Parallel and Distributed Computational Models, 2001.
32. Steering Committee Member, 13th International Conference on Parallel and Distributed Computing and Systems (PDCS), 2001.
33. Program Vice-chair for algorithms, International Symposium on Parallel Architectures, Algorithms, and Networks, 2002.
34. Advisory Committee Member, 14th International Conference on Parallel and Distributed Computing and Systems (PDCS), 2002.

35. Chair, Search Committee for Editor-in-Chief of IEEE Transactions on Computers, 2002.
36. Keynote speaker, Eighth International Conference, COCOON, 2002.
37. Keynote speaker, International Symposium on Parallel Architectures, Algorithms, and Networks, 2002.
38. Program Committee Chair, IEEE International High Performance Computing Conference, 2002.
39. Member, Advisory Committee, Software Track, Hawiian International Conference on System Sciences (HICSS36), 2003.
40. Publications Chair, IEEE Symposium on Applications and the Internet, SAINT, 2003.
41. Conference Chair, IASTED Conference on Computer Science and Technology, CST 2003-.
42. Keynote speaker, IEEE International Symposium on Computers and Communications, ISCC, 2003.
43. Keynote speaker, International Conference on Advanced Computing and Communications, ADCOM, 2003.
44. Member, Advisory Board, Labortory for Interdisciplinary Information Science and Technology, University of Central Florida, 2005-.
45. Keynote speaker, International Symposium on Parallel and Distributed Processing and Aplications, ISPA, 2005.
46. Co-chair, Steering Committee, International Symposium on Parallel and Distributed Processing and Aplications, ISPA, 2005.
47. Member, Advisory Board, International Conference on Innovations and Real-Time Applications of Distributed Sensor Networks, 2005-.
48. Co-Conference Chair, International Conference on Information Technology, Systems and Management.
49. Keynote Speaker, IEEE Sensor Networks, Ubiquitous, and Trustworthy Computing, 2006.
50. Steering Committee Member, IEEE International Conference on Computers and Communication, 2007-.
51. Conference co-chair, International Conference on Information Systems, Management and Technology, 2007-.
52. R. C. Bose Memorial Keynote Speaker, International Conference on Interdisciplinary Mathematical and Statistical Techniques, Shanghai, China, 2007.
53. Member, Advisory Board, International Conference on Technology, Communication and Education, IEEE, 2008.
54. Keynote Speaker, International Conference on Information Systems, Technology, and Management, Dubai, UAE, 2008.
55. Keynote Speaker, International Symposium on Advances in Computer and Sensor Networks, Zhengzhou, China, 2008.
56. Distinguished Lecture Series speaker at several universities.
57. External reviewer for several CS and CSE departments in the US and international.
58. Served on several NSF and NIH panels.
59. Served as session chair at several conferences.

60. Member of many conference program committees.
61. Reviewed papers for numerous journals and conferences.
62. Have given many invited presentations at national and international conferences and at universities and industrial organizations.

HONORS AND AWARDS

1. Colonel Ogilive medal 1965, First in All India Higher Secondary Exam (several thousand students).
2. Science Talent Search Scholarship, 1965.
3. President of India Gold Medal, First in class 1970, IIT/Kanpur, May 1970 (approx. 300 students).
4. Silver Medal, First in Electrical Engineering, IIT/Kanpur, May 1970 (approx. 80 students).
5. IBM Fellowship, Cornell University, 1970-1971.
6. Cornell University Fellowship, 1971-1973.
7. IEEE certificate of appreciation, 1982.
8. Outstanding Professor Award, Institute of Technology Student Board, University of Minnesota, 1986.
9. Senior Member, IEEE, August 1986.
10. Distinguished service award, International Conference on Parallel Processing, 1987.
11. Certificate of appreciation, 1987 Conference on expert systems technology in the ADP environment.
12. Fellow, Supercomputer Institute, University of Minnesota, 1985.
13. Fellow, IEEE, Jan. 1988. Citation: For contributions to computer algorithms, computer-aided design, and large-scale systems.
14. University of Minnesota Rochester Center for Continuing Education and Extension certificate for ``outstanding service and dedication in teaching", 1989.
15. Research achievement award, University of Florida, 1992.
16. IEEE meritorious service certificate, 1995.
17. Fellow, American Association for the Advancement of Science (AAAS), Oct. 1995. Citation: For contributions to the design and analysis of algorithms, parallel computing, and electronic computer aided design.
18. Teaching Incentive Program Award, University of Florida, 1995.
19. Fellow, Association for Computing Machinery (ACM), 1996. Citation: For contributions to data structures, design and analysis of algorithms, multiprocessor scheduling, electronic computer-aided design, and parallel computing.
20. Charter Member, IEEE Computer Society's Golden Core, 1996.
21. IEEE Computer Society Taylor L. Booth Education Award ``for contributions to Computer Science and Engineering education in the areas of data structures, algorithms, and parallel algorithms", 1997.
22. University of Florida Research Foundation Professorship, 1997-2000.
23. Distinguished Alumnus Award, Indian Institute of Technology, Kanpur, ``in recognition of his outstanding and seminal contributions in the field of Computer Science and Engineering", 2001.
24. Original Member, Highly Cited Researchers Database, 2002.

25. Member, European Academy of Sciences, 2002. Citation: For outstanding and lasting contributions to computer science and fundamental developments in the area of data structures and algorithms.
26. ACM recognition of service award, 2002.
27. IEEE Computer Society W. Wallace-McDowell Award, 2003. Citation: For contributions to the theory of NP-hard and NP-complete problems.
28. ACM Karl Karlstrom Outstanding Educator Award, 2003. Citation: For outstanding contributions to computing education through inspired teaching, development of courses and curricula for distance education, contributions to professional societies, and authoring significant textbooks in several areas including discrete mathematics, data structures, algorithms, and parallel and distributed computing.
29. *Best Paper Award*. X. Xu and S. Sahni, Approximation algorithms for sensor deployment. *Innovations and Real-Time Applications of Distributed Sensor Network (DSN) Symposium*, 2006.
30. *Best Paper Award*. S. Sahni, N. Rao, S. Ranka, Y. Li, E. Jung, and N. Kamath, Bandwidth scheduling and path computation algorithms for connection-oriented networks. *International Conference on Networking*, 2007.

GRANTS AND GIFTS

1. University of Minnesota Grant in Aid of Research for the Academic Year 1973-74
2. Education Development Program Grant 1974-75
3. National Science Foundation Research Grant DCR74-10081, Nov. 74-April 77
4. National Science Foundation Research Grant MCS76-21024, Nov. 76-April 79
5. National Science Foundation Research Grant MCS78-15455, Nov. 78-April 81
6. National Science Foundation Research Grant MCS80-05856, Sept. 80 - Feb. 84
7. Office of Naval Research Grant N00014-80-C-0650, July 80 - Dec. 1984.
8. MEIS research grant, 1983.
9. National Science Foundation Research Grant Oct 1983 - Oct 1986, $186,000.
10. MEIS Equipment Grant, 1983, $20,000.
11. MEIS DASE Grant, 1983, $125,000.
12. MEIS PACE Grant, 1983, $40,000.
13. National Science Foundation Equipment Grant 1984, $20,000.
14. National Science Foundation Equipment Grant 1984, $150,000 (includes matching funds from university sources).
15. CDC Research Grant, 1984, $35,000.
16. CDC Research grant, 1984, $95,000.
17. MEIS matching grant, 1984, $25,000.
18. NSF, CER grant, $3.8 million (including matching monies and supplements), 1985-1990 (Grant Manager: Sahni; PIs: Boley, Ibarra, Rosen, Sahni; plus four additional participating faculty).
19. Mentor Graphics CAD software donation, $982,300, 1987.
20. NSF research grant, $269,864, 1987-1990.
21. AT&T gift in support of CAD research, $15,000, 1988.
22. AT&T gift in support of CAD research, $10,000, 1991.

23. ACM/IEEE fellowship for graduate student support, $12,000, 1991.
24. SIGDA library award, $1000, 1991.
25. NSF research grant, $262,550, 1991-1994.
26. NSF, II/SS Grant (Grant Manager: Sahni, PIs: Chow, Ritter, Sahni, Su, Yau), approx. $1.3M (excludes matching funds from University sources). 1992-1997.
27. AT&T gift in support of CAD research, $15,000, 1992.
28. Army Research Office, research grant ``Parallel Algorithms'', $227,385, 1995-1998.
29. NIH, research grant ``Algorithms for compression and registration of brain MRI''(PI: Sahni, grant has four collaborators), $816,036, 1995-1998.
30. US Army, Waterways Experiment Station (PI: sahni, grant has 2 co-PIs), grant ``Framework for matching applications with parallel machine, $50,000, 1997-1998.
31. NSF, education grant ``Mainstreaming parallel and distributed computing in the Computer Science undergraduate curriculum'', (co-PI), $391,565, 1996-1999.
32. NSF, education grant ``Randomization techniques in the undergraduate and graduate curriculum'', (co-PI), $373,312, 1998-2001.
33. NIH, research grant ``Real-time dose computation and treatment planning''(PI: Sahni, grant has four collaborators), $1,377,809, 2000-2004.
34. NSF, research grant ``An algorithmic evaluation of optical interconnection networks,'' (co-PI), $285,000, 2000-2003.
35. Nortel Networks, research grant ``Efficient clustering and QoS routing in mobile ad-hoc networks,'' $18,900, 2001.
36. NSF, medium ITR research grant ``Information extraction from massive data sets'', $409,000, 2003-2009.
37. Intel Corp., research grant, ``Fault-tolerant grid computing,'' (with S. Ranka), $62,000, 2004-2005.
38. Pennsylvania State University, research grant ``Knowledge management,'' (with S. Ranka (PI), J. Hammer, C. Jermaine), $149,999, 2004-2005.
39. UT/Batelle, research grant ``Sensor deployment,'' $150,000, 2006-2008.
40. Advanced Algorithms and Systems, research grant ``Advanced scheduling of high-speed networks,'' (with S. Ranka), $52,000 (includes UF matching), 2006-2007.

The remaining sections of this chapter are organized around different ways to implement symbol tables. Different strategies are preferable given different assumptions. The first case considered is where the identifiers are known in advance and no deletions or insertions are allowed. Symbol tables with this property are called *static*. One solution is to sort the names and store them sequentially. Using either the binary search or Fibonacci search method of section 7.1 allows us to find any name in $O(\log_2 n)$ operations. If each name is to be searched for with equal probability then this solution, using an essentially balanced binary search tree, is optimal. When different identifiers are searched for with differing probabilities and these probabilities are known in advance this solution may not be the best. An elegant solution to this problem is given in section 9.1.

In section 9.2 we consider the use of binary trees when the identifiers are not known in advance. Sequential allocation is no longer desirable because of the need to insert new elements. The solution which is examined is AVL or height balanced trees. Finally in section 9.3 we examine the most practical of the techniques for dynamic search and insertion, hashing.

9.1 STATIC TREE TABLES

Definition: A *binary search tree T* is a binary tree; either it is empty or each node in the tree contains an identifier and:

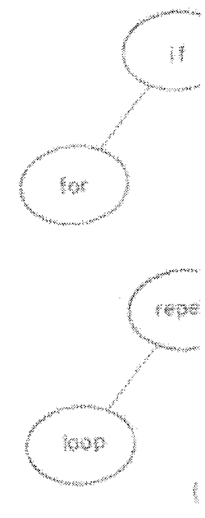
(i) all identifiers in the left subtree of T are less (numerically or alphabetically) than the identifier in the root node T;

(ii) all identifiers in the right subtree of T are greater than the identifier in the root node T;

(iii) the left and right subtrees of T are also binary search trees. For a given set of identifiers several binary search trees are possible. Figure 9.1 shows two possible binary search trees for a subset of the reserved words of SPARKS.

To determine whether an identifier X is present in a binary search tree, X is compared with the root. If X is less than the identifier in the root, then the search continues in the left subtree; if X equals the identifier in the root, the search terminates successfully; otherwise the search continues in the right subtree. This is formalized in algorithm SEARCH.

From Fundamentals of Data Structures by Horowitz and Sahni; Computer Science Press, 1976




```
procedure S
  //search
  LCHIL
  Otherw
  binary
  i ← T
  while i ≠
    case
      :X < I
      :X = I
      :X > I
    end
  end
end SEAR
```

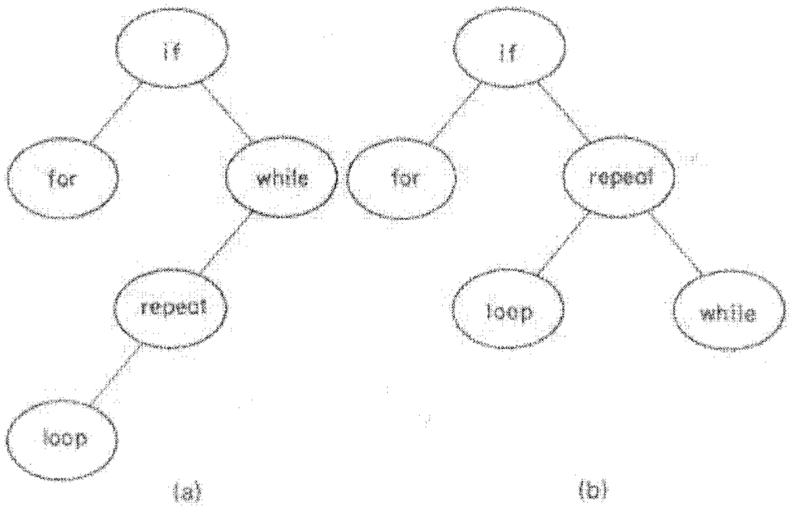


Figure 9.1 Two possible binary search trees

```
procedure SEARCH(T,X,i)
  //search the binary search tree T for X. Each node has fields
    LCHILD, IDENT, RCHILD. Return i = 0 if X is not in T.
    Otherwise, return i such that IDENT(i) = X. LCHILD(empty
    binary tree) = RCHILD(empty binary tree) = 0.//
  i ← T
  while i ≠ 0 do
    case
      :X < IDENT(i): i ← LCHILD(i)        //search left subtree//
      :X = IDENT(i): return
      :X > IDENT(i): i ← RCHILD(i)        //search right subtree//
    end
  end
end SEARCH
```

In our study of binary search in chapter 7, we saw that every sorted file corresponded to a binary search tree. For instance, a binary search on the file (**do**, **if**, **stop**) corresponds to using algorithm SEARCH on the binary search tree of figure 9.2. While this tree is a full binary tree, it need not be optimal over all binary search trees for this file when the identifiers are searched for with different probabilities. In order to determine an optimal binary search tree for a given static file,

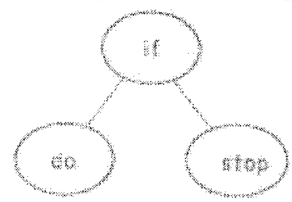


Figure 9.2 Binary search tree corresponding to a binary search on the file (**do**,**if**,**stop**).

we must first decide on a cost measure for search trees. In searching for an identifier at level k using algorithm SEARCH, k iterations of the **while** loop of lines 2–8 are made. Since this loop determines the cost of the search, it is reasonable to use the level number of a node as its cost.

Consider the two search trees of figure 9.1 as possibly representing the symbol table of the SPARKS compiler. As names are encountered a match is searched for in the tree. The second binary tree requires at most three comparisons to decide whether there is a match. The first binary tree may require four comparisons, since any identifier which alphabetically comes after **if** but precedes **repeat** will test four nodes. Thus, as far as worst case search time is concerned, this makes the

balanced binary search tree that mimics a binary search

while the table is being built and so it is difficult to achieve this ideal without making the time required to add new entries very high. This is so because in some cases it would be necessary to restructure the whole tree to accommodate the new entry and at the same time have a full binary search tree. It is, however, possible to keep the trees balanced so as to ensure both an average and worst case retrieval time of $O(\log n)$ for a tree with n nodes. We shall study one method of growing balanced binary trees. These balanced trees will have satisfactory search and insertion time properties.

Height Balanced Binary Trees

Adelson-Velskii and Landis in 1962 introduced a binary tree structure that is balanced with respect to the heights of subtrees. As a result of the balanced nature of this type of tree, dynamic retrievals can be performed in $O(\log n)$ time if the tree has n nodes in it. At the same time a new identifier can be entered or deleted from such a tree in time $O(\log n)$. The resulting tree remains height balanced. The tree structure introduced by them is given the name AVL-tree. As with binary trees it is natural to define AVL trees recursively.

Definition: An empty tree is height balanced. If T is a nonempty binary tree with T_L and T_R as its left and right subtrees, then T is *height balanced* iff (i) T_L and T_R are height balanced and (ii) $|h_L - h_R| \leq 1$ where h_L and h_R are the heights of T_L and T_R respectively.

The definition of a height balanced binary tree requires that every subtree also be height balanced. The binary tree of figure 9.7 is not height balanced since the height of the left subtree of the tree with root 'APRIL' is 0 while that of the right subtree is 2. The tree of figure 9.8 is height balanced while that of figure 9.9 is not. To illustrate the processes involved in maintaining a height balanced binary search tree, let us try to construct such a tree for the months of the year. This time let us assume that the insertions are made in the order MARCH, MAY, NOVEMBER, AUGUST, APRIL, JANUARY, DECEMBER, JULY, FEBRUARY, JUNE, OCTOBER and SEPTEMBER. Figure 9.10 shows the tree as it grows and the restructuring involved in keeping the tree balanced. The numbers within each node represent the difference in heights between the left and right subtrees of that node. This number is referred to as the balance factor of the node.

Definition: The *balance factor*, $BF(T)$, of a node T in a binary tree is defined to be $h_L - h_R$ where h_L and h_R are the heights of the left

Loading Order

As in the case of internal tables, the order in which key values are entered into the hash table is important. To the extent possible, an attempt should be made to enter these values in order of nonincreasing frequency of search. When this is done, new entries should be added to the end of the overflow chain rather than at the front.

10.2.3 Tree Indexing—*B*-Trees

The AVL tree of §9.2 provided a means to search, insert and delete entries from a table of size n using at most $O(\log n)$ time. Since these same functions are to be carried out in an index, one could conceivably use AVL trees for this application too. The AVL tree would itself reside on a disk. If nodes are retrieved from the disk, one at a time, then a search of an index with n entries would require at most 1.4 log n disk accesses (the maximum depth of an AVL tree is 1.4 log n). For an index with a million entries, this would mean about 23 accesses in the worst case. This is a lot worse than the cylinder sector index scheme of §10.2.1. In fact, we can do much better than 23 accesses by using a balanced tree based upon an m-way search tree rather than one based on a binary search tree (AVL trees are balanced binary search trees).

Definition: An *m-way search tree*, T, is a tree in which all nodes are of degree $\leq m$. If T is empty, (i.e., $T = 0$) then T is an m-way search tree. When T is not empty it has the following properties:

(i) T is a node of the type

$$n, A_0, (K_1, A_1), (K_2, A_2), \ldots, (K_n, A_n)$$

where the A_i, $0 \leq i \leq n$ are pointers to the subtrees of T and the K_i, $1 \leq i \leq n$ are key values.

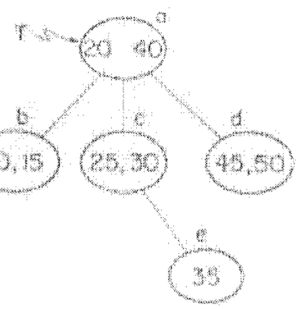
(ii) $K_i < K_{i+1}$, $1 \leq i < n$

(iii) All key values in the subtree A_i are less than the key value K_{i+1}, $0 \leq i < n$

(iv) All key values in the subtree A_n are greater than K_n.

(v) The subtrees A_i, $0 \leq i \leq n$ are also m-way search trees.

As an example of a 3-way search tree consider the tree of figure 10.9 for key values 10, 15, 20, 25, 30, 35, 40, 45 and 50. One may easily verify that it satisfies all the requirements of a 3-way search



node	schematic format
a	2,b,(20,c),(40,d)
b	2,0,(10,0),(15,0)
c	2,0,(25,0),(30,e)
d	2,0,(45,0),(50,0)
e	1,0,(35,0)

Figure 10.9 Example of a 3-way search tree.

tree. In order to search for any key value X in this tree, we first "look into" the root node $T = a$ and determine the value of i for which $K_i \le X < K_{i+1}$ (for convenience we use $K_0 = [-\infty]$ and $K_{n+1} = [+\infty]$ where $[-\infty]$ is smaller than all legal key values and $[+\infty]$ is larger than all legal key values). In case $X = K_i$ then the search is complete. If $X \ne K_i$ then by the definition of an m-way search tree X must be in subtree A_i if it is in the tree. When n (the number of keys in a node) is "large," the search for the appropriate value of i above may be carried out using binary search. For "small" n a sequential search is more appropriate. In the example if $X = 35$ then a search in the root node indicates that the appropriate subtree to be searched is the one with root $A_1 = c$. A search of this root node indicates that the next node to search is e. The key value 35 is found in this node and the search terminates. If this search tree resides on a disk then the search for $X = 35$ would require accessing the nodes a, c and e for a total of 3 disk accesses. This is the maximum number of accesses needed for a search in the tree of figure 10.9. The best binary search tree for this set of key values requires 4 disk accesses in the worst case. One such binary tree is shown in figure 10.10.

Algorithm MSEARCH searches an m-way search tree T for key value X using the scheme described above. In practice, when the search tree

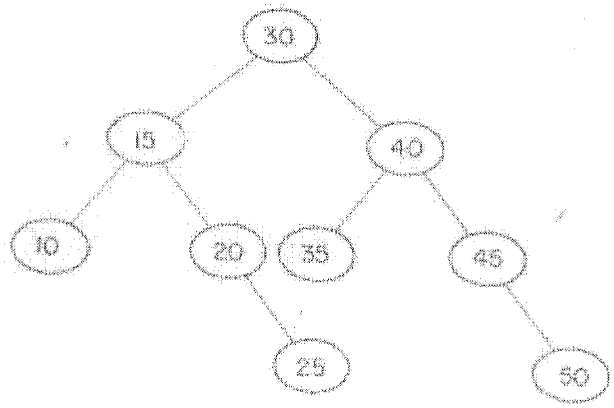


Figure 10.10 Best AVL-tree for data of figure 10.9

represents an index, the tuples (K_i, A_i) in each of the nodes will really be 3-tuples (K_i, A_i, B_i) where B_i is the address in the file of the record with key K_i. This address would consist of the cylinder and surface numbers of the track to be accessed to retrieve this record (for some disks this address may also include the sector number). The A_i, $0 \le i \le n$ are the addresses of root nodes of subtrees. Since these nodes are also on a disk, the A_i are cylinder and surface numbers.

```
   procedure MSEARCH(T,X)
   //Search the m-way search tree T residing on disk for the key
     value X. Individual node format is n,A_0,(K_1,A_1), ...,(K_n,A_n),
     n < m. A triple (P,i,j) is returned. j = 1 implies X is found
     at node P, key K_i. Else j = 0 and P is the node into which
     X can be inserted//
1    P <- T; K_0 <- [-∞]; Q <- 0          //Q is the parent of P//
2    while P ≠ 0 do
3      input node P from disk
4      Let P define n, A_0, (K_1,A_1), ..., (K_n,A_n)
5      K_{n+1} <- [+∞]
6      Let i be such that K_i ≤ X < K_{i+1}
7      if X = K_i then [//X has been found// return (P,i,1)]
8      Q <- P; P <- A_i
9    end
     //X not in T; return node into which insertion can take place//
10   return (Q,i,0)
11 end MSEARCH
```

Analyzing algorithm MSEARCH is fairly straightforward. The maximum number of disk accesses made is equal to the height of the tree T. Since individual disk accesses are very expensive relative to the time needed to process a node (i.e. determine the next node to access, lines 4–8) we are concerned with minimizing the number of accesses needed to carry out a search. This is equivalent to minimizing the height of the search tree. In a tree of degree m and height $h \geq 1$ the maximum number of nodes is $\sum_{0 \leq i \leq h-1} m^i = (m^h - 1)/(m - 1)$. Since each node has at most $m - 1$ keys, the maximum number of entries in an m-way tree index of height h would be $m^h - 1$. For a binary tree with $h = 3$ this figure is 7. For a 200-way tree with $h = 3$ we have $m^h - 1 = 8 \times 10^6 - 1$.

Clearly, the potentials of high order search trees are much greater than those of low order search trees. To achieve a performance close to that of the best m-way search trees for a given number of entries n, it is necessary that the search tree be balanced. The particular variety of balanced m-way search trees we shall consider here is known as a B-tree. In defining a B-tree it is convenient to reintroduce the concept of failure nodes as used for optimal binary search trees in §9.1. A failure node represents a node which can be reached during a search only if the value X being searched for is not in the tree. Every subtree with root $= 0$ is a point that is reached during the search iff X is not in the tree. For convenience, these empty subtrees will be replaced by hypothetical nodes called failure nodes. These nodes will be drawn square and marked with an F. The actual tree structure does not contain any such nodes but only the value 0 where such a node occurs. Figure 10.11 shows the 3-way search tree of figure 10.9 with failure nodes. Failure nodes are the only nodes that have no children.

Definition: A B-tree, T, of order m is an m-way search tree that is either empty or is of height ≥ 1 and satisfies the following properties:

(i) the root node has at least 2 children

(ii) all nodes other than the root node and failure nodes have at least $\lceil m/2 \rceil$ children

(iii) all failure nodes are at the same level.

The 3-way search tree of figure 10.11 is not a B-tree since it has failure nodes at levels 3 and 4. This violates requirement (iii). One possible B-tree of order 3 for the data of figure 10.9 is shown in figure 10.12. Notice that all nonfailure nodes are either of degree 2 or 3. In fact, for a B-tree of order 3, requirements (i), (ii) and the definition

balanced m-way search tree that permits search, insert, delete in $O(\log_m n)$ time